RICE UNIVERSITY

Probabilistic Models for Genetic and Genomic Data with Missing Information

by

Stephanie Carinne Hicks

A Thesis Submitted
in Partial Fulfillment of the
Requirements for the Degree

Doctor of Philosophy

Approved, Thesis Committee:

Marek Kimmel, Chair and Thesis
Co-Director
Professor of Statistics

Sharon E. Plon, Thesis Co-Director
Professor of Pediatrics and Molecular and
Human Genetics, Baylor College of
Medicine

James R. Thompson
Professor of Statistics

Luay Nakhleh
Associate Professor of Computer Science

Houston, Texas

April, 2013

ABSTRACT

Probabilistic Models for Genetic and Genomic Data with Missing Information

by

Stephanie Carinne Hicks

Genetic and genomic data often contain unobservable or missing information. Applications of probabilistic models such as mixture models and hidden Markov models (HMMs) have been widely used since the 1960s to make inference on unobserved information using some observed information demonstrating the versatility and importance of these models. Biological applications of mixture models include gene expression data, meta-analysis, disease mapping, epidemiology and pharmacology and applications of HMMs include gene finding, linkage analysis, phylogenetic analysis and identifying regions of identity-by-descent. An important statistical and informatics challenge posed by modern genetics is to understand the functional consequences of genetic variation and its relation to phenotypic variation. In the analysis of whole-exome sequencing data, predicting the impact of missense mutations on protein function is an important factor in identifying and determining the clinical importance of disease susceptibility mutations in the absence of independent data determining impact on disease. In addition to the interpretation, identifying co-inherited regions of related individuals with Mendelian disorders can further narrow the search for disease susceptibility mutations. In this thesis, we develop two probabilistic models in application of genetic and genomic data with missing information: 1) a mixture model to estimate a posterior probability of functionality of missense mutations and

2) a HMM to identify co-inherited regions in the exomes of related individuals. The first application combines functional predictions from available computational or *in silico* methods which often have a high degree of disagreement leading to conflicting results for the user to assess the pathogenic impact of missense mutations on protein function. The second application considers extensions of a first-order HMM to include conditional emission probabilities varying as a function of minor allele frequency and a second-order dependence structure between observed variant calls. We apply these models to whole-exome sequencing data and show how these models can be used to identify disease susceptibility mutations. As disease-gene identification projects increasingly use next-generation sequencing, the probabilistic models developed in this thesis help identify and associate relevant disease-causing mutations with human disorders. The purpose of this thesis is to demonstrate that probabilistic models can contribute to more accurate and dependable inference based on genetic and genomic data with missing information.

Acknowledgements

I would like to thank everyone who was involved with my graduate career at Rice University. First, I would like to thank my committee members and specifically my advisors Marek Kimmel and Sharon Plon for their mentoring, guidance and patience. I have been so fortunate to be mentored by two fantastic advisors who have taught me how to think critically and strive for my very best. To my past and present officemates, I have loved the fun and thought-provoking conversations over the years. To Javier Rojo who first introduced me to Rice by selecting me for RUSIS many years ago. Thank you to all the faculty, staff and students in our department who have provided encouragement and support. To my collaborators at Baylor College of Medicine (Hannah Cheung, Bradford Powell, David Wheeler) and my collaborators at Rice (Navin Rustagi and Roberto Bertolusso) thank you for all of your help.

Next, I would like to thank my family and friends who have supported me on this long road. My parents, Mark and Marika, who have taught me you are never too young or too old to find and pursue your dreams in life. Your love and support have provided the strength to get where I am today. To my sister, Vanessa, for always listening and reminding me to have fun in life. To Chris, thank you for eternal optimism and loving support, especially during these last few weeks of thesis writing. My grandparents, Doug, Linda and extended family who have supplied countless prayers and kind words.

Finally, thank you to the selfless educators who dedicated their lives to inspire young people to pursue their dreams. To Mr. Burgess and Mrs. Tatum for igniting my interested in mathematics in middle school and high school. To Lisa for meeting me by chance and Dr. Warner, Lisa, Karin and Monica for allowing me to join the HHMI and LA-STEM research scholars program at LSU and supporting me to pursue a PhD.

Contents

Illustrations

Tables

Chapter 1

Introduction

An important statistical and informatics challenge posed by modern genetics is to understand the functional consequences of genetic variation and its relation to phenotypic variation. For example, in cancer genomics this broadly translates to identifying cancer susceptibility mutations which can impact decisions related to prevention, prognosis and treatment in the affected patient and at-risk family members. This has been a difficult task as human cancer is immensely complex and an increase in the risk cancer susceptibility can result from combinations and permutations of hundreds of genetic alterations.

Probabilistic models such as mixture models and hidden Markov models (HMMs) are two examples of powerful statistical models which have been successfully applied in a vast area of biological (and non biological) research. Both models make inference on observable and unobservable (or latent) data which is often the case when using genetic or genomic data. Mixture models are mixtures of probability distributions, but the information about which subpopulation each observation belongs to is typically missing. These models were first introduced by Karl Pearson [Pearson, 1894] and have been applied in a wide range of areas such as biology, economics, astronomy, engineering. A hidden Markov model can be considered a generalization of a mixture model where the hidden variables (or latent variables), which control the mixture component to be selected for each observation, are related through a Markov process rather than independent of each other. Hidden Markov models are used to model an

underlying, unobservable process given observations emitted by the process. They have been applied since the late 1960s and early 1970s because of the rich mathematical structure that has been developed for the models [Rabiner, 1989]. The large number of applications, including gene finding, linkage analysis, phylogenetic analysis and identifying regions of identity-by-descent, indicate the versatility and importance of these models.

In the advent of next-generation sequencing, obtaining data is no longer the bottleneck in genomics, but rather the bottleneck is the interpretation of the genetic alterations. In this thesis, probabilistic models in application for genetic and genomic data with missing information will be considered: 1) mixture model to estimate the probability of functionality of missense mutations using observed predictions from known bioinformatics or *in silico* algorithms and 2) a HMM to identify co-inherited regions in the exomes (the part of the genome formed by exons which are the coding portions of genes) of related individuals.

In Chapter 1, a brief introduction to probabilistic models for data with missing information, specifically mixture models and hidden Markov models, will be given followed by a brief discussion of examples and applications. Chapter 2 begins with the mathematical theory, methodology and specific properties of mixture models (Section 2.1), Expectation-Maximization algorithm (Section 2.2) and hidden Markov models (Section 2.3). Section 2.4 conveys basic principals in genetic mapping such as Mendelian genetics and recombination. The genetic architecture of diseases is briefly discussed in Section 2.5. The last two sections (2.6, 2.7) in this chapter motivate the applications of mixture models and hidden Markov models for genetic and genomic data formally developed in Chapter 4 and Chapter 5.

In Chapter 3, a study is given of how predictions of missense mutation functional-

ity depend on the *in silico* method and sequence alignment employed. We show there is a high degree of disagreement between the predictions of functionality produced by these *in silico* methods. The material discussed in this chapter was published in two articles in Human Mutation [Hicks et al., 2011, Hicks et al., 2013] and the former was highlighted and discussed in Nature [Baker, 2012].

In Chapter 4, we develop a new method for interpreting the functionality of missense mutations. In Section 4.1 we discuss previously proposed solutions for combining predictions of functionality and investigate the level of agreement between these *in silico* algorithms. In Section 4.2, we develop two statistical models based on the capture-recapture paradigm which combine often discordant functional predictions in a statistically rigorous manner and estimate a unified posterior probability for each mutation being deleterious. Unlike previous methods, our approach, referred to as postMUT and postMUT (simple), requires no training set or calibration and estimates the sensitivity and specificity of each individual *in silico* method in the absence of a gold standard by taking advantage of the fact these methods disagree. Several applications of these models are considered in Section 4.3. When a gold standard is available, we show the sensitivity and specificity estimates using the postMUT models closely match the sensitivity and specificity estimated directly using the known functional mutation status. As another application, we use two matched normal/tumor breast cancer genomes and show an enrichment of deleterious mutations using mutations found only in the tumor and only in the normal compared to mutations both in the normal and tumor. These posterior probabilities may be used as a filter when inferring the functionality of missense mutations in exome-scale sequencing projects. In Chapter 5, we investigate the performance of the postMUT (simple) and postMUT models by performing simulation studies. We assess bias and mean squared

error (MSE) as a function of the number of mutations in a given dataset.

In Chapter 6, we consider several inhomogeneous HMMs used to predict regions of identity-by-descent (IBD) in siblings affected by an autosomal recessive disease using the identity-by-state (IBS) status observed from whole-exome sequencing data. To improve accuracy, we extend a previously developed first-order HMM by exploring conditional emission probabilities and a second-order dependence structure between observed variants calls. The models are formally discussed in Section 6.3, 6.4 and 6.5. Additionally, we show the conditional emissions vary as a function of minor allele frequency. The models are evaluated on simulated exome sequencing data and real human exome sequencing data to identify regions of IBD in Section 6.6 and 6.7. These models provide researchers a tool to filter large portions of the exome in search of finding the causal variant for a given disease. We conclude with Chapter 7 and discuss ideas for future directions.

1.1 Motivation for Probabilistic Models

1.1.1 Mixture Models

Mixture models are probabilistic models representing mixtures of distributions (or subpopulations). Observations are considered to be drawn independently from any subpopulation, but we do not observe which subpopulation it was drawn from. Mixture models were first discussed by Karl Pearson [Pearson, 1894] and they have been a widely studied topic since the late 1960s and 1970s [McLachlan and Peel, 2000].

Consider a random sample $\mathbf{X} = (X_1, \ldots, X_n)$ where each observation is identically and independently drawn from the probability distribution $f(x)$. Each independent observation is drawn from one of the g finite subpopulations. Let $\mathbf{Y} = (Y_1, \ldots, Y_n)$

represent what subpopulation the ith observation was drawn from. Each Y_i is a categorical variable taking values in $(1, \ldots, g)$. A mixture model is a weighted sum of g component densities $f_j(x)$

$$f(x) = \sum_{j=1}^{g} p_j f_j(x)$$

where p_j are the nonnegative mixture proportions or weights which must sum to one.

$$0 \leq p_j \leq 1 \qquad \text{and} \qquad \sum_{j=1}^{g} \pi_j = 1$$

ffInstead of using the likelihood directly, it is common to use the logarithm of the likelihood of $\mathbf{X}$ which is given by

$$l(\theta) = \log L(\theta|\mathbf{x}) = \sum_{i=1}^{n} \log \sum_{j=1}^{g} p_j f_j(x_i)$$

Note, we can also re-write $\mathbf{Y}$ as $\mathbf{Y} = (\mathbf{Y}_1, \ldots, \mathbf{Y}_n)$ where each $\mathbf{Y}_i$ is a g-dimensional vector of indicator variables depending on whether jth component of $\mathbf{Y}_i$ (i.e. $Y_{ij} = (\mathbf{Y}_i)_j$ is equal to 1 or 0. In this case, $\mathbf{Y}_i$ follows a a multinomial distribution with probabilities $p_1, \ldots p_g$.

There are three basic questions of when applying mixture models:

1. Are the number of components or subpopulations g considered fixed and pre-specified before estimation or is g fixed but not necessarily known and should be inferred from the data?

2. Are the component densities $f_j(x_i)$ defined parametric, semi-parameteric or non-parametric?

3. Given the observations $\mathbf{X}$, how do we estimate the weight proportions p_j and, if applicable, the parameters θ in the parametric component densities $f_j(x_i|\theta)$?

These questions have all been widely studied and various solutions to all have been provided. In this thesis we assume g is fixed and pre-specified before parameter estimation. We assume our component densities $f_j(x_i|\theta)$ to be parametrized by θ. Finally, parameter estimation is performed using the Expectation-Maximization algorithm [Dempster et al., 1977].

1.1.2 Markov Models

Instead of observations being identically and independently drawn from a population such as in Section 1.1.1, sometimes a dependence between observations is assumed. If the past and present information is known about a particular process, then the "Markov" property states that inference on the future depends only on the present information. Hidden Markov models use the set of probabilistic models called Markov processes or Markov chains that allow the probability observing a particular state be dependent on the previous states. If the process is in continuous-time, then it is called a Markov process or if it is in discrete-time then it is called a Markov chain. The Markov model is characterized by a transition probability matrix P which describes the probability of the process or chain moving from state to state.

Consider a first-order discrete-time **Markov chain** $\{X_n,\ n = 0, 1, 2, \ldots\}$ with distribution P and finite state space $S = \{s_0, s_1, \ldots, s_N\}$. The first-order Markov property states that given that process is in state s_i there is a given fixed probability P_{ij} that it will next be in state s_j which only depends on the previous state (e.g. first-order):

$$P_{ij} = P[X_{n+1} = s_j | X_n = s_i]$$

such that

$$P_{ij} \geq 0, \quad i,j \geq 0; \qquad \sum_{j=0}^{\infty} P_{ij} = 1, \quad i = 0, 1, \ldots$$

and the joint probability of the chain with length L is given by

$$\begin{aligned} P(X) &= P[X_L, X_{L-1}, \ldots, X_1] \\ &= P[X_L|X_{L-1}, \ldots, X_1]P[X_{L-1}|X_{L-2}, \ldots, X_1] \ldots P[X_1] \\ &= P[X_L|X_{L-1}]P[X_{L-1}|X_{L-2}] \ldots P[X_1] \quad \text{(Markov property)} \\ &= P[X_1] \prod_{n=2}^{L} P[X_n|X_{n-1}] \end{aligned}$$

A first-order continuous-time **Markov process** $\{X(t) : t \geq 0\}$ with a finite state space S is a stochastic process satisfying the Markov property

$$P[X(t+h) = j|X(t) = i] = q_{ij}h + o(h), \qquad i \neq j$$

where q_{ij} is the i,j entry of the transition intensity matrix Q where

$$q_{ii} = -\sum_{j \neq i} q_{ij}$$

1.1.3 Hidden Markov Models

A hidden Markov model can be considered a generalization of a mixture model where the hidden variables (or latent variables), which control the mixture component to be selected for each observation, are related through a Markov process rather than independent of each other. These models arise when the chain of interest is unobservable, but a different set states are observable called emissions conditional on the unobservable state. To formalize the notation, let the unobservable Markov chain be given by $\{\pi_n, n = 0, 1, 2, \ldots\}$ and is characterized by the transition probabilities

$$P_{ij} = P[\pi_n = j|\pi_{n-1} = i]$$

The beginning of the process is defined by the stationary distribution $a_i = P(\pi_1 = i)$ or probability of starting in state i. We do not observe the path of the chain π_n, but we do observe a process $\{X_n, n = 0, 1, 2, \ldots\}$ with a set of states and given emission probabilities

$$e_j(b) = P[X_n = b|\pi_n = j]$$

defined as the probability that we observe state b given the unobservable chain is in state j. Then, the joint probability of the observed sequence X and unobservable sequence π is given by

$$P(X, \pi) = a_i \prod_{n=1}^{L} e_{\pi_n}(x_n) P_{\pi_n, \pi_{n-1}}$$

There are three basic questions of interest when applying HMMs:

1. Given the observed sequence X and model parameters $\theta = \{P, e, a\}$, how do we predict most probable path π^* or predict which state the unobservable chain π is in given the observable states?

2. Given the observed sequence X and model parameters θ, how do we compute $P(X|\theta)$ or the probability of the observed sequence given the model parameters?

3. When the model parameters θ are unknown, how do we estimate the parameters that maximize $P(X|\theta)$?

These three questions have all been answered using the following algorithms:

1. If the goal is to predict which state the unobservable chain π is in given the observable states, then we find the most probable path π^* by computing

$$\pi^* = \underset{\pi}{\text{argmax}}\ P(X, \pi)$$

This can be computed using the Viterbi algorithm [Rabiner, 1989, Durbin et al., 1998].

2. The probability of the observed sequence X can be found by marginalizing over the joint probability by summing over all possible hidden paths of π:

$$P(X) = \sum_{\pi} P(X, \pi)$$

This can be computed using the Forward algorithm [Rabiner, 1989, Durbin et al., 1998].

3. If the transition probabilities and emission probabilities are unknown, then they can be estimated using the Baum-Welch algorithm [Rabiner, 1989, Durbin et al., 1998]. The Baum-Welch algorithm (also called the forward-backward algorithm) is an EM algorithm commonly used to estimate the parameters in a hidden Markov model. For details related to the estimation see [Blimes, 1998].

1.2 Biological Applications

1.2.1 Mixure Models

Karl Pearson [Pearson, 1894] was the first to use mixture models on a dataset measuring the ratio of forehead to body length of $n = 100$ crabs [Weldon, 1893] indicating the evolution of two new subspecies. At that time, parameter estimation was performed using method of moments [Lehmann and Casella, 1998], but in the late 1960s maximum-likelihood based approaches were introduced, in particular the Expectation-Maximization algorithm [Dempster et al., 1977] which significantly increased the use of mixture models.

Over the past 40-50 years, mixture models have been successfully applied in a diverse range of fields. In medical applications, mixture models have been commonly employed in gene expression data, disease mapping, drug development, and meta-analyses [Schlattmann, 2009].

In the analysis of gene expression data, mixture models are used to find differences in gene expression levels between subgroups of individuals [Lee et al., 2000, Efron et al., 2001, McLachlan et al., 2006]. The most basic model assumes a two-component mixture model where the gene is either being differentially expressed or nondifferentially expressed. Ultimately, this could lead to individualized therapy on the basis of gene expression in the tumor cells such as breast cancer [Brennan et al., 2007].

Disease mapping or investigating the geographical distribution of disease occurrence through the use of mixture models has been studied for over three decades [Schlattmann and Böhning, 1993] in applications such as malaria [Rattanasiri et al., 2004] and breast cancer [Schlattmann, 2000]. In meta-analyses, covariate-adjusted mixture models have been introduced in metaregression [Schlattmann, 2000] where the heterogeneity between studies is explained by covariates in finite mixture models. This is beneficial because it alleviates the assumption of a normal distribution of the random effects in the usual random effects model. For the analysis of microarrays, bayesian mixture models have been frequently used in meta-analyses [Conlon, 2008].

Many software packages and tools have been implemented to ease the computational burden. Within R [R Core Team, 2012], there are over 50 packages listed related to cluster analysis and finite mixture models under the CRAN task view (http://cran.r-project.org/web/views/Cluster.html). Both maximum-likelihood and bayesian approaches have been widely implemented in various R packages. Some popular mixture models include mixtools [Benaglia et al., 2009] and mclust [Fraley and Raftery, 2012]. Other programming languages which have implemented tools for mixture models include MATLAB, SAS, Python. A few commonly used standalone free packages for estimation of mixture models include DispapWin for disease mapping [Schlattmann, 1996] and EMMIX for the fitting of mixtures of normal and t-components [McLachlan et al., 1999].

Other non-biological applications of mixture models include speaker identification [Reynolds and Rose, 1995], image analysis [Permuter et al., 2003], and economics [Brigo and Mercurio, 2002].

1.2.2 Hidden Markov Models

The most successful applications of hidden Markov models in computational biology and bioinformatics have been profile HMMs and HMM-based gene finders [Eddy, 1998]. Profile HMMs were introduced in 1994 [Krogh et al., 1994a] allowing a multiple sequence alignment to be converted into a position-specific scoring system ultimately allowing databases to be searched for homologous sequences. In general, each column in the sequence alignment is modeled as a 'match state', 'insert state', both having 4 or 20 nucleotide or amino acid emission probabilities each, and a 'delete state', with no emission probabilities. Several profile HMM software packages exist: SAM [Hughey and Krogh, 1996, Karplus et al., 1998], HMMER [Eddy, 1998], PFTOOLS [Bucher et al., 1996], and HHMPro [Baldi et al., 1994]. Examples of profile HMM libraries include PROSITE profiles [Sigrist et al., 2010] and Pfam [Finn et al., 2010].

Gene finding or gene prediction is defined as identifying regions of genomic DNA that encode genes from a given sequence. The two main types of gene finders are *ab initio* and homology-based (or similarity-based). A third class of hybrids also exists. Given the observed nucleotide sequence, the HMM assigns 'classes' to each position such as exons, introns, Poly(A) tails, and TATA boxes [Knapp and Chen, 2007] which identifies genes. Many HMM-based gene finders have been developed with varying levels of accuracy. A review of existing methods [Mathé et al., 2002, Wang et al., 2004] and a comparison of six contemporary methods [Knapp and Chen, 2007] (Augus-

tus, Genezilla, GenomeScan, GlimmerHMM, SNAP and Twinscan) have been performed. Applications of HMM-based gene finders have existed since the early 1990s [Krogh et al., 1994b, Kulp et al., 1996, Burge and Karlin, 1997, Henderson et al., 1997, Krogh, 1997, Lukashin and Borodovsky, 1998].

CpG islands [Bird, 1987] are made up of a few hundred to a few thousand repeat CG dinucleotides. Identifying CpG islands in a given stretch of genomic sequence is biologically important because it can be an indication of the promoter region of a gene, which in turn helps locate genes across a long stretch genomic sequence. Given an observed CG dinucleotide, the hidden states are defined as island states and non-island states where the nucleotides C_+ and G_+ represent the islands states and C_- and G_- represent the non-islands states. Hidden Markov models have been applied to identify the switching between islands and non-islands states [Durbin et al., 1998].

Another area that HMMs have been successfully applied is the prediction of protein secondary structure. Predictions from a single protein sequence [Asai et al., 1993, Stultz et al., 1993, Goldman et al., 1996] are based on the idea that certain types of amino acids were associated with certain secondary structure environments [Balding et al., 2007]. Likelihoods associated with HMMs of protein structure have been applied to DNA sequence data [Churchill, 1989] and are explained in detail [Thorne et al., 1996].

Other biological applications of HMMs include pairwise sequence alignments [Durbin et al., 1998], phylogenetic analysis [Felsenstein, 1981, Felsenstein and Churchill, 1996, Thorne et al., 1996], genetic linkage mapping [Lander and Green, 1987, Kruglyak et al., 1996], identifying regions of identity-by-descent which will be discussed in Chapter 2. Non-biological applications include speech recognition [Ramesh and Wilpon, 1992] and [Watson and Chung Tsoi, 1992, Rabiner, 1989, Zelinka and Sigmund, 2010], ecology

[Guttorp, 1995], image analysis [Romberg et al., 2001], economics [Hamilton, 1989, Albert and Chib, 1993] and music analysis [Qi et al., 2007].

Chapter 2

Background

2.1 Mixture Models

In Chapter 1, we introduced the idea of mixture models. In this section, we will formalize the notation. Suppose we have a random sample $\mathbf{X} = (X_1, \ldots, X_n)$ independently and identically distributed $f(x|\theta)$ and parameterized by θ. The the likelihood $L(\theta|\mathbf{X})$ is interpreted as a function of the parameters where the data is fixed.

$$L(\theta|\mathbf{X}) = \prod_{i=1}^{n} f(x_i|\theta)$$

In the maximum likelihood framework, our goal is to find the θ which maximizes L or find

$$\theta^* = \underset{\theta}{\operatorname{argmax}}\ L(\theta|\mathbf{X})$$

To find the maximum likelihood estimate (MLE), we compute the gradient of the log likelihood $\log(L(\theta|\mathbf{X}))$, set equal to 0 and solve for θ. Let $\hat{\theta}$ be the MLE of θ.

When we are not able to observe the entire data set because it contains missing values or is incomplete, then other statistical estimation techniques besides maximum-likelihood esimateion should be employed. In Chapter 1, we previously considered a random sample $\mathbf{X} = (X_1, \ldots, X_n)$ where each observation is identically and independently drawn from the probability distribution $f(x)$. Let $f(x)$ be a parametric probability distribution $f(x|\theta)$. Each independent observation is drawn from one of the g finite subpopulations. Let $\mathbf{Y} = (Y_1, \ldots, Y_n)$ represent what subpopulation the

ith observation was drawn from. Each Y_i is a categorical variable taking values in $1, \ldots, g$. Therefore, if we assume X_i is drawn from the jth subpopulation (i.e. $Y_i = j$) with probability distribution $f_j(x|\theta)$, we can write

$$f(x|\theta) = \sum_{j=1}^{g} p_j f_j(x|\theta)$$

where p_j are the nonnegative mixture proportions or weights which must sum to one:

$$0 \leq p_j \leq 1 \qquad \text{and} \qquad \sum_{j=1}^{g} \pi_j = 1$$

Because we do not observe which subpopulation the ith observation was drawn from, we cannot compute the MLE directly.

2.2 Expectation-Maximization Algorithm

The Expectation-Maximization algorithm is a method of finding the maximum-likelihood estimate of the parameters of an underlying distribution from a given data set when the data is incomplete or has missing values [Blimes, 1998]. Let $\mathbf{X} = (X_1, \ldots, X_n)$ be the *incomplete* data or observed data. Let $\mathbf{Y} = (Y_1, \ldots, Y_n)$ be the *augmented* data we do not observe because it is missing. We define $\mathbf{Z} = (\mathbf{X}, \mathbf{Y})$ as the *complete* data and

$$f(\mathbf{z}|\theta) = f(\mathbf{x}, \mathbf{y}|\theta) = f(\mathbf{x}|\mathbf{y}, \theta) f(\mathbf{y}|\theta)$$

and the marginal distribution of $\mathbf{X}$ is given by

$$f(\mathbf{x}|\theta) = \int f(\mathbf{x}, \mathbf{y}|\theta) d\mathbf{y}$$

We define $L(\theta|\mathbf{x}) = f(\mathbf{x}|\theta)$ as the *incomplete-data likelihood* and $L(\theta|\mathbf{x}, \mathbf{y}) = f(\mathbf{x}, \mathbf{y}|\theta)$ as the *complete-data likelihood*. Technically speaking, $\mathbf{Y}$ is considered unknown and random and we can think of $L(\theta|\mathbf{x}, \mathbf{Y})$ as a function of a random variable $\mathbf{Y}$ where $\mathbf{x}$ is constant.

In the EM algorithm, we want to maximize $L(\theta|\mathbf{x}, \mathbf{Y})$ in θ, but we do this using a conditional expectation

$$Q(\theta|\theta^{(t)}) = E_{\mathbf{Y}|\mathbf{x},\theta^{(t)}}[\log L(\theta|\mathbf{x}, \mathbf{Y})]$$

where $\theta^{(t)}$ is the parameter estimate for θ at the previous tth iteration which we use to evaluate the expectation. The EM algorithm moves in iterations between two steps: The Expectation Step (E-Step) and the Maximization Step (M-Step).

2.2.1 Estimation Steps

E-Step

Take the expectation of the complete-data likelihood with respect to $\mathbf{Y}$ conditional on the observed data $\mathbf{x}$ and the current parameter estimates $\theta^{(t)}$

$$Q(\theta|\theta^{(t)}) = E_{\mathbf{Y}|\mathbf{x},\theta^{(t)}}[\log L(\theta|\mathbf{x}, \mathbf{Y})] = \int \log f(\mathbf{y}, \mathbf{x}|\theta) f(\mathbf{y}|\mathbf{x}, \theta^{(t)}) d\mathbf{y}$$

where $f(\mathbf{y}|\mathbf{x}, \theta^{(t)})$ is the marginal distribution of the unobserved or missing data [Dempster et al., 1977, Blimes, 1998].

M-Step

Find the new $\theta^{(t+1)}$ that maximizes $Q(\theta|\theta^{(t)})$

$$\theta^{(t+1)} = \underset{\theta}{\operatorname{argmax}}\ Q(\theta|\theta^{(t)})$$

where θ represents the set of parameters we are searching for to maximize the likelihood and $\theta^{(t)}$ are the estimated parameters at the previous iteration and will be used to evaluate the expectation [Dempster et al., 1977, Blimes, 1998].

2.2.2 Monotonicity of EM algorithm

As the EM algorithm iterates between the E-Step and M-Step, the goal is to maximize $L(\theta|\mathbf{x})$ by working with only $L(\theta|\mathbf{x},\mathbf{y})$ and $f(\mathbf{y}|\mathbf{x},\theta^{(t)})$. With an initial guess $\theta^{(0)}$, the EM algorithm will 'improve' and converge to a local maximum of $\log L(\theta|\mathbf{x})$ [Mengersen et al., 2011] or

$$\log[f(\mathbf{x}|\theta^{(t+1)})] \geq \log[f(\mathbf{x}|\theta^{(t)})]$$

2.2.3 Confidence Intervals using EM Algorithm

When all the data is complete and does not contain any missing data, we can calculate a $100(1-\alpha)\%$ confidence interval for θ by computing the Fisher Information matrix [Shao, 2003] from the sample and

$$[\hat{\theta} - Z_{\alpha/2}(\frac{1}{\sqrt{I(\theta)}}), \hat{\theta} + Z_{\alpha/2}(\frac{1}{\sqrt{I(\theta)}})]$$

When there is missing information, the EM algorithm is commonly used to estimate the parameters. To calculate the confidence intervals for MLEs of incomplete data out of the EM algorithm we use the observed information matrix of the incomplete data [Louis, 1982, Meilijson, 1989, Lange, 1995, Oakes, 1999]. Louis (1982) defines the notation of the gradient and the negative of the second derivatives of the complete likelihood,

$$S(Y,\theta) = \frac{\partial \log L(\theta|Y)}{\partial \theta} \quad \text{and} \quad B(Y,\theta) = -\frac{\partial^2 \log L(\theta|Y)}{\partial \theta^2}$$

and the gradient of the observed likelihood

$$S^*(X,\theta) = \frac{\partial \log L(\theta|X)}{\partial \theta}$$

where $S^*(x,\theta) = E_{Y|X,\theta}[S(Y,\theta)]$ and $S^*(x,\hat{\theta}) = 0$. Then, the observed information matrix of the incomplete data can be obtained using

$$I_X(\theta) = E_{Y|X,\theta}[B(Y,\theta)] - E_{Y|X,\theta}[S(Y,\theta)S^T(Y,\theta)] + S^*(x,\theta)S^{*T}(x,\theta)$$

or another way to think about it is

$$I_X = I(\hat{\theta}) = I_Y(\theta) - I_{Y|X}$$

Oakes (1999) shows the function $Q(\theta|\theta^{(t)})$ can be used in the maximization of the observed likelihood $L(\theta|x)$. Therefore, when calculating the observed information matrix of the incomplete data, it is sufficient to use

$$I(\theta) = -\frac{\partial^2 Q}{\partial \theta^2}|_{\theta=\hat{\theta}}$$

To calculate a $100(1-\alpha)\%$ confidence interval for θ, we then use the same formula as above

$$[\hat{\theta} - Z_{\alpha/2}(\frac{1}{\sqrt{I(\theta)}}), \hat{\theta} + Z_{\alpha/2}(\frac{1}{\sqrt{I(\theta)}})]$$

2.3 Hidden Markov Models

Hidden Markov models are an extension of mixture models imposing a dependence structure in the data. Several good introductions to HMMs exist [Rabiner, 1989, Durbin et al., 1998]. In this section, the mathematical theory behind different properties of HMMs will be discussed.

2.3.1 Higher-order HMM

The first-order Markov chain can be extended to an n-th order Markov chain. Consider the case when $n = 2$. A second-order Markov chain $\{X_n, n = 0, 1, 2, \ldots\}$ with

distribution P and finite state space $S = \{s_0, s_1, \ldots, s_N\}$ has the property

$$P_{ijk} = P[X_{n+1} = k | X_n = j, X_{n-1} = i]$$

A second-order hidden Markov model is similar to the first-order except the difference is the way the transition probability matrix is defined. The unobservable chain given by π is now characterized by the transition probabilities

$$P_{ijk} = P[\pi_n = k | \pi_{n-1} = j, \pi_{n-2} = i]$$

Let the beginning of the process be defined by $a_i = P(\pi_1 = i)$ or probability of starting in state i. We do not observe the path of the chain π_n, but we do observe a new set of states with given second-order emission probabilities

$$e_{ij}(b) = P[X_n = b | \pi_n = j, \pi_{n-1} = i]$$

defined as the probability that we observe state b given the unobservable chain is in state j at the n step and state i at $n-1$. Then, the joint probability of the observed sequence X and unobservable sequence π is given by

$$P(X, \pi) = a_{x_1} P_{\pi_1, \pi_2} \prod_{n=2}^{L} e_{\pi_{n-1}\pi_n}(x_n) P_{\pi_n, \pi_{n-1}, \pi_{n-2}}$$

The same three questions asked above have also been answered with second-order extensions of the following algorithms: Viterbi algorithm [Thede and Harper, 1999], Forward algorithm and Baum-Welch [Watson and Chung Tsoi, 1992]. The Baum-Welch algorithm (also called the forward-backward algorithm) is an EM algorithm commonly used to estimate the parameters in a hidden Markov model. For details related to the estimation steps see [Blimes, 1998].

2.3.2 Stationary and Non-Stationary HMM

A non-stationary Markov model $\{X_n, n = 0, 1, 2, \ldots\}$ with distribution P is characterized by the property that the transition probability matrix is a function of the state duration distribution $P_i(d)$ where d is the state duration or often referred to as sojourn time. For stationary HMMs, if d is the duration of a particular state s_k then the duration distribution is given by

$$P_k(d) = P_{kk}^{d-1}(1 - P_{kk})$$

and is geometrically distributed [Djuric and Chun, 2002] for Markov chains. The duration distribution is exponentially distributed in continuous-time Markov processes. In some applications, this assumption of a geometric distribution is inappropriate. Non-stationary HMMs overcome this limitation and allow the transition probabilities to depend on the state duration

$$P_{ij}(d) = P(\pi_n = j | \pi_{n-1} = \pi_{n-2} = \ldots = \pi_{n-d} = i)$$

The relationship between the duration distribution $P_k(d)$ and transition probabilities $P_{ij}(d)$ are given by

$$P_{ii}(d) = \begin{cases} 1 - P_i(d) & \text{if } d = 1 \\ \frac{1-\sum_{k=1}^{d} P_i(k)}{1-\sum_{l=1}^{d-1} P_i(l)} & \text{if } d > 1 \end{cases}$$

and for $i \neq j$

$$P_{ij}(d) = \begin{cases} w_{ij} P_i(d) & \text{if } d = 1 \\ w_{ij} \frac{P_i(d)}{1-\sum_{l=1}^{d-1} P_i(l)} & \text{if } d > 1 \end{cases}$$

with weights satisfying

$$\sum_j w_{ij} = 1 \qquad \text{for } i \neq j$$

An important relationship between HMMs and NSHMMs is that if $d = 1$ or $P_{ij}(d)$ is constant where $i \neq j$ then the NSHMM reduces to a stationary HMM

[Bae et al., 2008]. The duration distribution $P_i(d)$ may be defined by any distribution, but it is often defined as a truncated Poisson distribution

$$P_i(d) \propto \frac{\epsilon_i^d e^{-\epsilon_i}}{d!}$$

where ϵ_i is the parameter for the Poisson distribution associated with the ith state and $\epsilon > 0$ [Djuric and Chun, 2002, Bae et al., 2008].

2.3.3 Homogeneous and Inhomogeneous HMM

An inhomogeneous Markov model $\{X_n, n = 0, 1, 2, \ldots\}$ with distribution P is characterized by the property that the probability of transitioning between states depends not only on the previous and current state, but also at what position the process is along a given sequence. For a homogeneous Markov model the transition probability matrix is the same regardless of position.

2.3.4 Bayesian Methods for HMM

The parameters in hidden Markov models are most commonly estimated using iterative algorithms as mentioned in Section 1.1.3 such as the Viterbi, Forward, and Baum-Welch algorithms. These iterative algorithms may not be the most efficient for parameter estimation. Other techniques such as Markov Chain Monte Carlo (MCMC) estimation can be used. A review of how to implement HMMs for the Bayesian modeler has been performed [Scott, 2002]. Scott discusses how to obtain the posterior distribution of the parameters, to estimate the HMM, and to use diagnostics to asses model validity and MCMC convergence.

2.4 Basic Principles in Genetic Mapping

2.4.1 Mendelian Inheritance

Gregor Mendel studied heritable traits in the 1860s and described two laws that explain how heritable traits are passed from parents to offspring. The first law called Law of Segregation says a pair of genes for a given trait will segregate randomly into gametes which are passed on to the offspring. The second law called Law of Independent Assortment says that during gamete formation the segregation of one gene pair is independent of other gene pairs [Balding et al., 2007]. Therefore, a Mendelian trait or disorder is described as a trait that is influenced by a single gene and follows a Mendelian inheritance pattern.

2.4.2 Recombination

The second law (Law of Independent Assortment) is true for many pairs of genes but fails when considering genes that are linked. The statistic that describes the level of genetic linkage between a pair of genes is $\theta = P(\text{recombinate gamete})$:

$$\theta = \begin{cases} 1/2 & \text{when genes are unlinked} \\ 0 & \text{when genes are completely linked} \\ (0, 1/2) & \text{genes are said to be "linked" or "in genetic linkage"} \end{cases}$$

where recombination is described as the chromosomal exchange of DNA. Recombination frequencies are not distributed evenly across chromosomes because there are clear hotspots of recombination.

Genetic distances between genes can be estimated using recombination fractions and genetic map functions. The Haldane map function [Haldane, 1919] is described

using the recombination frequency θ and map distance d in centiMorgans (cM)

$$\theta = \frac{1}{2}(1 - e^{-2d})$$

or the inverse estimates the genetic distance in terms of recombination frequency

$$d = -\frac{1}{2}\log(1 - 2\theta)$$

Other genetic map functions have been proposed [Kosambi, 1944, Carter and Falconer, 1951, Sturt, 1976, Rao et al., 1977, Felsenstein, 1979, Karlin and Liberman, 1978]

2.5 Genetic Architecture of Diseases

Genetic diseases have been characterized in the following two ways: 1) Mendelian diseases and 2) common diseases. We have previously discussed Mendelian genetics in Section 2.4.1. Mendelian diseases are influenced by a single causative mutation (low frequency) or single gene and follow the Mendelian inheritance pattern.

The second type of disease, common diseases, are considered to be complex diseases in the sense that multiple mutations of higher population frequency or multiple genes and possibly their interaction with each other and the environment all have an effect on the disease outcome. There is a debate about the genetic basis of complex diseases such as cancer and diabetes. Two proposed etiologies for complex traits are:

1. Common Disease, Common Variant (CDCV) hypothesis: common variants with low to modest effect sizes that influence the disease in a significant portion of the population [Reich and Lander, 2001, International HapMap Consortium, 2003, Risch and Merikangas, 1996], or the

2. Common Disease, Rare Variant (CDRV) hypothesis: rare variants with large effect sizes that individually only account for a small proportion of disease in

the population [Pritchard, 2001, Bodmer and Bonilla, 2008, Schork et al., 2009, Manolio et al., 2009]

Evidence for both hypotheses has been given [Schork et al., 2009]. Figure 2.1 depicts a graph showing the feasibility of identifying variants by risk allele frequency and strength of genetic effect (odds ratio). This figure was taken from Manolio et al. (2009). Figure 2.2 represents the genetic architecture of disease and what current methods are applied to finding these risk and causative alleles. This figure was taken from Singleton et al. (2010).

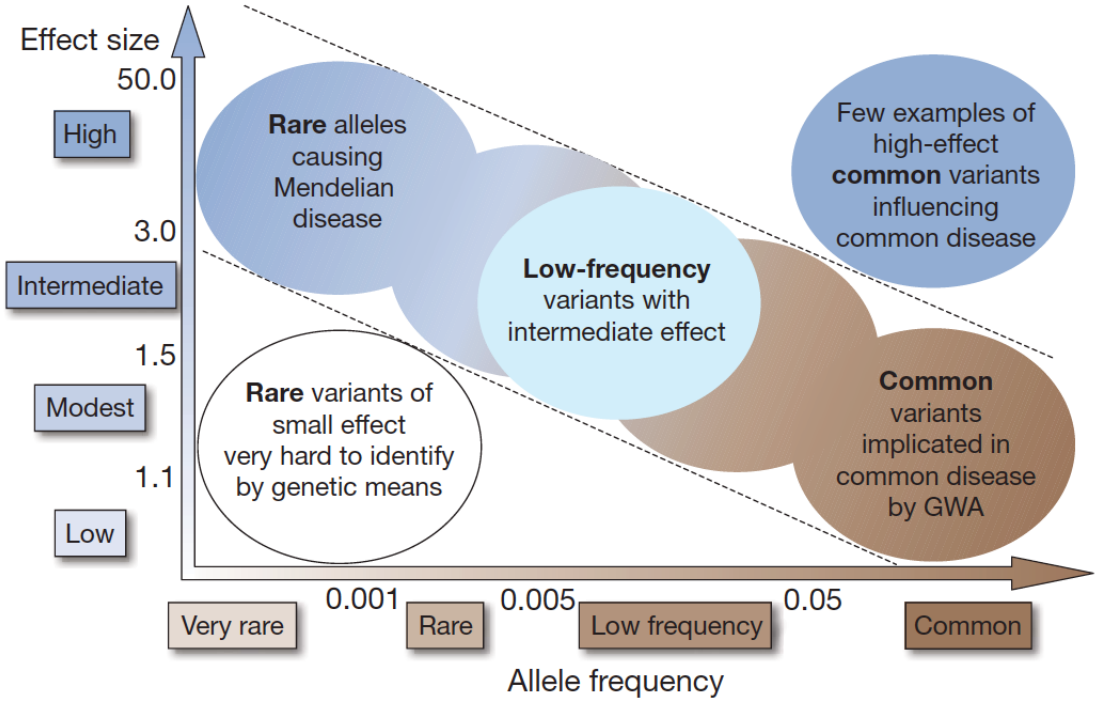

Figure 2.1 : Feasibility of identifying variants by allele frequency and strength of genetic effect [Manolio et al., 2009]

Genetic studies typically use a particular method to target a particular range of allele frequencies and effect size believed to be associated or linked to the trait of

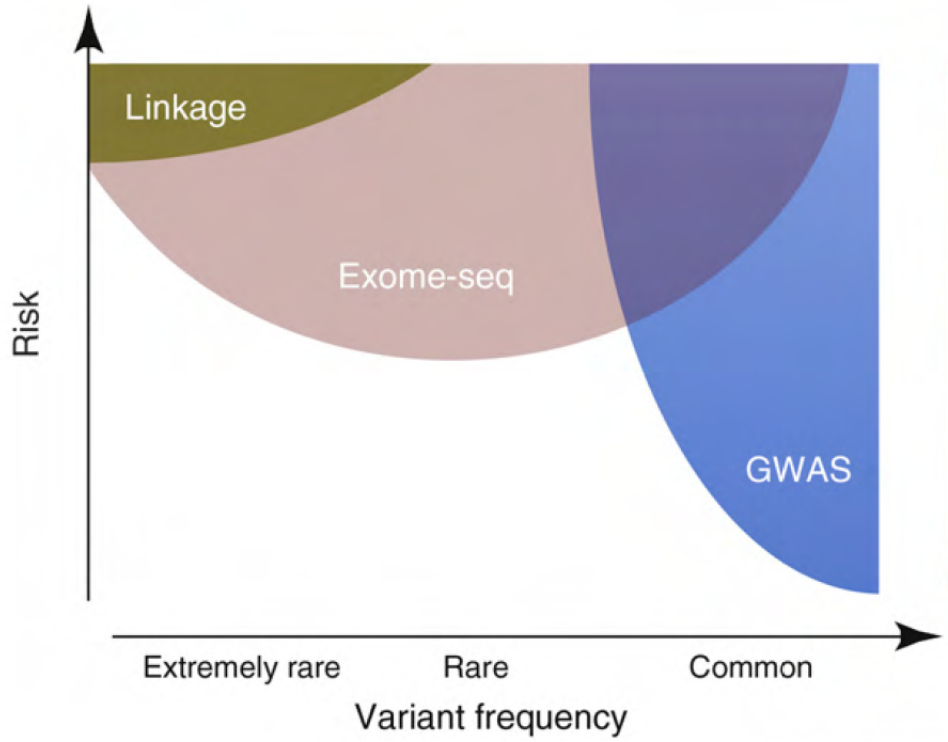

Figure 2.2 : The landscape of genetic architecture of disease and the current methods to finding the risk and causative variants [Singleton et al., 2010]

interest. Genome-wide association studies (GWAS) have focused on common alleles with modest effect sizes while linkage studies which require large pedigrees have focused on rare alleles with large effect sizes such as Mendelian traits. In the advent of next-generation sequencing, exome sequencing can allow low-frequency or rare, moderate-risk alleles to be found. Current methods available to identify these rare variants are whole genome and whole exome sequencing technologies. The first proof-of-concept example was by Ng et al. (2009) who showed that next-generation sequencing can be used to identify disease-causing mutations in only twelve unrelated individuals (four cases and eight controls) with a rare dominantly inherited disorder, Freeman-Sheldon syndrome, with a previously known cause [Ng et al., 2009,

Cirulli and Goldstein, 2010]. Another important example is Ng et al. 2010 which was the first paper to identify the gene for a rare Mendelian disorder (Miller syndrome) with an unknown cause using exome sequencing in only four cases and eight controls [Ng et al., 2010]. These successes in identifying causes of Mendelian diseases indicate that whole genome sequencing of an even smaller number of individuals can identify causal variants.

2.6 Missense Mutations and Mixture Models

Nonsynonymous changes found as single nucleotide polymorphisms (SNPs) resulting in amino acid substitutions in the encoded protein product are called missense mutations and may or may not affect protein function. Predicting the impact of missense mutations on protein function is an important factor in identifying and determining the clinical importance of disease-causing mutations. Many computational or *in silico* methods including SIFT [Ng and Henikoff, 2001], Align-GVGD [Mathe et al., 2006, Tavtigian et al., 2008b], PolyPhen-2 [Ramensky et al., 2002, Adzhubei et al., 2010], and Xvar [Reva et al., 2007, Reva et al., 2011] have been developed, but they often lead to conflicting results leaving the user without guidance in assessing the pathogenic impact of missense mutations on protein function as discussed in Chapter 3 of this thesis.

In Chapter 4, we develop two statistical models based on the capture-recapture paradigm which combine the discordant functional predictions in a statistically rigorous manner and estimate a unified posterior probability of functionality or pathogenicity for each missense mutation. Unlike previous methods, our probabilistic approach requires no training set or calibration and estimates the accuracy (sensitivity and specificity) of each individual *in silico* method in the absence of a gold standard by

taking advantage of the fact these methods disagree. We define a mixture model with the weight parameter representing the overall proportion of deleterious mutations and perform parameter estimation using the Expectation-Maximization algorithm. Our approach is able to account for the accuracy of each *in silico* method when combining the functional predictions.

2.7 Identity-by-Descent and Hidden Markov Models

Several disease-gene identification projects have recently used whole-exome sequencing as a technique to identify disease-causing variants in unrelated individuals affected by the same Mendelian disorder [Choi et al., 2009, Ng et al., 2009, Ng et al., 2010]. These projects used an "intersection" approach [Robinson et al., 2011] by searching for the same rare variants found in all of the affected individuals while filtering or eliminating common variants found in large databases such as dbSNP, HapMap and 1000 Genomes Project. Though this is an effective technique in identifying genes associated with Mendelian disorders in unrelated individuals, it not ideal for related individuals because it does not take into account additional information from families. Instead, researchers have combined linkage analysis with whole-exome sequencing as a way to identify disease genes in related individuals; however, this technique is not optimized for the high error rates in whole-exome sequencing data [Rödelsperger et al., 2011]. Therefore, researchers studying Mendelian disorders are utilizing the theory of identity-by-descent as a tool to filter error-prone exome sequencing data and identify disease-causing mutations in areas of the genome that were transmitted through Mendelian inheritance only, in particular for autosomal recessive diseases.

In autosomal recessive disorders, all affected children share two haplotypes that

are *identical-by-descent* (IBD). Therefore the disease gene must be located in this IBD region [Rödelsperger et al., 2011]. In consanguineous families, chromosomal segments inherited IBD are genetic regions in two or more individuals that are inherited from a common ancestor while regions inherited *identical-by-state* (IBS) have identical DNA sequence but were not necessarily inherited from a common ancestor. In non-consanguineous families, the chromosomal segments can be defined IBD if each affected sibling inherits the same haplotype from each parent. In whole-exome sequencing, the observed genotypes of affected children are IBS or not for each ith marker. The goal is to predict the number of unobserved shared alleles IBD = 0, 1, 2 between affected siblings. Hidden Markov models are one way to make such predictions. These models are used to identify regions of IBD and are interested in predicting whether a given marker is IBD given the IBS status. Therefore, using the notation defined in Section 1.1.3, the IBD status is the unobservable markov chain π_n and the observable IBS status is given by X_n.

In Chapter 6 of this thesis, we consider several inhomogeneous hidden Markov models to identify IBD regions. These models are an extension from the first-order hidden Markov model developed by Rodelsperger et al. (2011) which was based on inhomogeneous transition probabilities. We also redefine the emission probabilities to conditional emission probabilities and develop a Viterbi-type algorithm for parameter estimation. These methods will provide researchers a tool to filter large portions of the exome in search of finding causal variants for a given disease.

Because the idea of employing HMMs to identify regions of IBD has been widely studied, a brief review of the methods for related individuals will be given.

2.7.1 Methods for related individuals

Leutenegger et al. (2003) used a first-order HMM to estimate the inbreeding coefficient while accounting for marker dependencies to ultimately make inference on the IBD status at each marker for a given inbred individual. Using a maximum-likelihood approach, their method defined transition probabilities that depend on the distance d_n between the n and $n+1$ markers in centimorgans (cM), δ the inbreeding coefficient and η the number of meiotic steps to the most recent common ancestor. The emission probabilities, which connect the observed genotypes with the unobserved hidden states, were defined in terms of population frequencies [Leutenegger et al., 2003]. Wang et al. (2006) expanded on this model by accounting for linkage disequilibrium (LD). Because they used marker data with a higher density, they could no longer assume markers to be in linkage equilibrium like Leutenegger et al. (2003). Instead, they assumed the nearby markers to be in linkage disequilibrium and argued applying the original model without accounting for LD would result in an over-prediction of regions of IBD. The main modification to the original method was to redefine emission probabilities as the probability of a genotype X_n being dependent not just on the IBD status π_n, but also the previous genotype X_{n-1} (or $P(X_n|X_{n-1}, \pi_n, \pi_{n-1})$). This second-order emission probability uses haplotype frequencies at two neighboring markers and also incorporates the probability of a genotyping error as well as missing data [Wang et al., 2006]. Though the emission probabilities were defined as second-order, the Markov chain still remained a first-order allowing the regular Viterbi algorithm to be used.

Albrechtsen et al. (2009) also employed a first-order HMM to estimate the probability of being IBD between pairs of related or distantly related individuals for each marker, but they redefined the hidden states at each marker to be defined as the

number of alleles shared IBD between pairs of individuals (i.e. 0, 1, 2) instead of the binary status of IBD or not IBD as before. Since they used a higher density marker data set, their model also accounted for linkage disequilibrium using haplotype frequencies, but the main difference between this method and the method of Wang et al. (2006) is that the marker genotypes in the emission probabilities were allowed to be conditioned on any of the previous markers, not just the adjacent one while assuming the same hidden state for the two markers (or $P(G_i^{j,k}|G_h^{j,k}, \pi_i = \pi_h)$). Han and Abney (2011) also developed a set of conditional emission probabilities which depended not only on allele frequency, but also haplotype frequency estimated using the observed and true genotypes. This method was developed for dense genotype data and the purpose was to incorporate LD in IBD estimations [Han and Abney, 2011].

A novel HMM approach was developed by Li et al. (2010) who introduced the concept of an inheritance-generating function between a pair of alleles in a specified pedigree structure. The hidden states represent the number of shared IBD alleles between each pair of individuals. This model is extended from Lander and Green (1987) in the sense that the marginal probability of being IBD at a particular marker depends not only on the observed IBS status, but also the relationship between the pair of individuals of interest. Because the derived transition probabilities must account for any type of relationship between the individuals, the inheritance-generating function was developed.

The most recent first-order HMM method is from Rodelsperger et al. (2011) who were the first to use whole-exome sequencing data while the three methods above all used a sequencing technology called SNP Affymetrix array. This is a platform to directly test for known polymorphisms across the genome of sizes ranging from 100K-2000K depending on the chip. This method defines the IBD status as a binary

indicator if the n affected siblings from a consanguineous or non-consanguineous families all share the observed genotype at a marker or not. The transition probabilities are inhomogeneous because they depend on position-specific recombination rates and the marginal probabilities, describing the probability of the hidden IBD state given the observed IBS state, are homogeneous. This method is specific for autosomal recessive Mendelian inheritance patterns. Another major difference is that the above methods incorporate observed genotypes into their transition probability matrices and emission probabilities, while this method only asks if the genotypes are equal or not. Thus, the emission probabilities are defined in terms of the probability of a false genotype call at a single variant position ϵ. Chapter 6 expands on the Rodelsperger et al. (2011) method by extending the model to inhomogeneous second-order HMMs and considers conditional emission probabilities which vary as a function of minor allele frequency.

Chapter 3

Interpreting the Functionality of Missense Mutations

A major bottleneck in genomics and bioinformatics is the interpretation of missense mutations. In the protein-coding regions of the genome, predicting the impact of missense mutations on protein function is a difficult task because missense mutations do not necessarily impact protein function. Many computational or *in silico* algorithms have been developed to predict the functionality or pathogenicity of mutations observed in human exomes, but they often lead to conflicting results leaving the researcher without guidance in how to prioritize the mutations identified for further evaluation in biological assays. These *in silico* methods base their predictions on the idea of using evolutionary conservation as a measure of pathogenicity by employing protein sequence alignments as the standard input to these *in silico* methods.

In this chapter, we perform a study investigating how functional predictions vary between using different *in silico* methods, and also vary using different protein sequence alignments. This study highlights the difficulty in predicting missense mutation functionality and shows the *in silico* methods have a high degree of disagreement. We note that Section 3.1 essentially contains the same information as the paper we published in Human Mutation [Hicks et al., 2011] which was recently highlighted and discussed in Nature [Baker, 2012]. Then, in Section 3.2 we review possible reasons for the disagreement between predictions of functionality and provide an example of the degree of disagreement. In Section 3.3 we discuss additional technical problems

related to using these *in silico* methods. Also, we give an example of the importance of properly assessing the accuracy of these methods which we published in Human Mutation [Hicks et al., 2013].

3.1 *in silico* Methods: Predictions of Functionality

A number of algorithms have been developed to predict the impact of missense mutations on protein structure and function including sequence and structure-based approaches. A review of available computational methods for assessing the functional effects of missense mutations has been performed [Ng and Henikoff, 2006, Karchin, 2009, Thusberg and Vihinen, 2009, Jordan et al., 2010]. Many methods base their predictions on phylogenetic information implying the pathogenicity of missense mutations is assessed from the observed amino acid variation at a given residue in the multiple sequence alignment employed. Variables that affect the prediction accuracy of these algorithms include the gene examined, the number of sequences in the alignment, the evolutionary distances among species, the algorithm used, and the importance of absolute amino acid conservation versus relatively conservative missense changes [Greenblatt et al., 2003]. Researchers have used multiple methods as a way to increase confidence in identifying deleterious mutations when predictive results differ between methods [Chan et al., 2007, Chun and Fay, 2009] but it has been argued these algorithms have major similarities underneath the lid and the correlation of their outputs is the result of similarity of their inputs, which is not a cause for increased confidence [Karchin, 2009]. Problems in comparing multiple methods extend further because there is no standard classification system used to categorize the predicted functionality of the variants, needed to provide a statistical measure of performance of the methods. Researchers have addressed this problem by grouping the predictions from

the algorithms into two main categories: variants that are predicted to be deleterious or neutral [Chan et al., 2007].

Several studies have compared the prediction accuracy of sequence [Balasubramanian et al., 2005, Mathe et al., 2006] and structure-based algorithms [Bao and Cui, 2005, Chan et al., 2007, Chao et al., 2008] using alignments generated by the algorithms or manually curated alignments. However, it remains unclear how the predictions change when the sequence alignment provided changes, since it has been suggested that when the outputs from the algorithms differ, it is most likely due to employing different protein sequence alignments [Karchin, 2009]. Past research has shown high predictive values for methods that use evolutionary sequence conservation, surprisingly with or without protein structural information [Chan et al., 2007]. Since sequence alignments influence sequence-based methods which ultimately generate measures of pathogenicity [Kryukov et al., 2007], it is important to determine which types of sequence alignments will lead to better in silico assessments [Tavtigian et al., 2008a].

In this study we employed four commonly used *in silico* algorithms:

1. **SIFT** (http://sift.jcvi.org/). The method Sorts Intolerant From Tolerant is a sequence homology-based tool that predicts variants in the query sequence as neutral or deleterious using normalized probabilities calculated from the input multiple sequence alignment [Ng and Henikoff, 2001]. SIFT obtains this multiple sequence alignment by internally generating it or by allowing the user to submit their own FASTA-formatted alignment. The alignment built by SIFT contains homologous sequences with a medium conservation measure of 3.0 where conservation is represented by information content [Schneider et al., 1986] to minimize false positive and false negative error. The authors mention better results may be obtained using only ortholog sequences because including par-

alogs can confound predictions at residues conserved only among the orthologs [Ng and Henikoff, 2002]. Variants at a position with normalized probabilities less than 0.05 are predicted deleterious and predicted neutral with a probability greater than or equal to 0.05.

2. **Align-GVGD** (http://agvgd.iarc.fr/). This method predicts variants in the query sequence based on a combination of Grantham Variation (GV), which measures the amount of observed biochemical evolutionary variation at a particular position in the alignment, and Grantham Deviation (GD), which measures the biochemical difference between the reference and amino acid encoded by the variant [Mathe et al., 2006]. The original classifier uses a set of five criteria based on GV and GD which classifies variants as neutral, unclassified' or deleterious [Mathe et al., 2006]. For example, in the extreme case of GV = 0, the alignment is completely conserved at that position and any other variant will be considered deleterious. The new classifier provides ordered grades ranging from the most pathogenic to least likely pathogenic [Tavtigian et al., 2008a]. The algorithm has primarily been used for a few clinically relevant tumor suppressor genes such as BRCA1, TP53 and the author provides highly manually curated alignments which may cause a favorable bias towards this algorithm when applied to the class of genes studied here. These alignments contain a small number of full-length ortholog sequences with a long range of evolutionary depth. The author argues the alignment should not only be restricted to true orthologs due to the biological phenomenon of functional diversification among paralogs [Abkevich et al., 2003], but also should sample enough sequences at sufficient evolutionary distance from each other (alignment depth) for the best accuracy of the algorithms [Tavtigian et al., 2008a]. The experimentalist can

provide his or her own alignment for other genes.

3. **PolyPhen-2** (http://genetics.bwh.harvard.edu/pph2/). This method is the latest tool developed by the authors of the original PolyPhen [Ramensky et al., 2002]. Its novel features include the set of predictive features, the alignment pipeline and the probabilistic classifier based on machine-learning methods. PolyPhen-2 predicts variants as benign, possibly damaging or probably damaging based on eight sequenced-based and three structure-based predictive features which were selected by an iterative greedy algorithm. Another useful feature is the algorithm calculates a Bayes posterior probability that a given mutation is deleterious [Adzhubei et al., 2010]. The web-based version requires use of the built in alignment pipeline, but the user may download and install the latest version of PolyPhen-2 to submit their own alignment. The alignment pipeline used in PolyPhen-2 selects homologous sequences using a clustering algorithm and then constructs and refines the alignment yielding an alignment containing both orthologs and paralogs that may or may not be full length, which yields a wider breadth of sequences but decreased depth compared with the Align-GVGD alignment. The authors argue this leads to more accurate predictions because a majority of deleterious variants affect protein structure compared to specific protein function [Adzhubei et al., 2010].

4. **Xvar** (http://xvar.org/). This recently developed web-based algorithm [Reva et al., 2011] by the same authors of the original algorithm combinatorial entropy optimization [Reva et al., 2007] cannot accept user-defined multiple sequence alignments from the investigator as input. The Xvar server can map the variant to both the Uniprot (http://www.uniprot.org/) and NCBI Reference Se-

quence (Refseq) protein (http://www.ncbi.nlm.nih.gov/refseq/) and to the 3D structure in Protein Data Bank (PDB) (http://www.pdb.org/pdb/) if available. Once the Uniprot IDs are identified, they are used to build local sequence alignments and extract information about the domain boundaries, annotated functional regions and protein-protein interaction instead of using full-length sequences as in the other three algorithms. Xvar predicts variants as neutral, low, medium or high.

3.1.1 Protein Sequence Alignments

The four multiple protein sequence alignments used as input to compare the variability of predictions from each algorithm include:

1. An automatically generated alignment from SIFT

2. An automatically generated alignment from PolyPhen-2 with a wide breadth of sequences (http://genetics.bwh.harvard.edu/pph2/). After downloading and installing the latest version of PolyPhen-2, the alignment automatically generated by the alignment pipeline can be obtained in a FASTA-formatted alignment.

3. A small highly curated alignment with long evolutionary depth ideal for Align-GVGD (http://agvgd.iarc.fr/). The BRCA1, MSH2, MLH1 and TP53 curated protein alignments can be directly obtained from the website and formatted into a FASTA-formatted alignment.

4. An uncurated alignment (http://www.uniprot.org/) automatically generated in Uniprot using the built in ClustalW feature with sequences included based on a criteria of 50% identity. This alignment was used as an unbiased alignment in the sense that none of the programs were built or trained on this type of

alignment, which makes it suitable for testing the variability of the predictions from these algorithms.

Specific details related to the settings for each *in silico* method are given here:

1. **SIFT**. The web-based method SIFT version 4.0.3 was used with all default settings. The alignment built by SIFT was created using the SIFT Sequence option. The other three multiple sequence alignments were each submitted under the SIFT Aligned Sequences option. Variants were predicted as neutral or deleterious.

2. **Align-GVGD**. The web-based method Align-GVGD was used with all default settings. The Align-GVGD alignment for BRCA1, MSH2, MLH1 and TP53 are freely available on the website. Each of the other three types of multiple sequence alignments were also submitted. Variants were predicted as neutral, unclassified or deleterious. For this study, the variants predicted as neutral and unclassified were grouped together as neutral variants.

3. **PolyPhen-2**. The latest version of PolyPhen-2 version 2.0.22 and helper programs were downloaded and installed on 3.06 GHz Intel Core 2 Duo processor with 6GB L2 Cache memory computer at Rice University. The standard output reported by the downloaded algorithm uses the default classifier model HumDiv, but predictions may also be obtained using the HumVar model reporting only minor differences between the two models (data not shown). The alignment built by PolyPhen-2 was automatically generated. The other three multiple sequence alignments could be submitted because the downloaded algorithm allows the user to submit their own FASTA-formatted alignment. Variants were predicted as benign, possibly damaging or probably damaging. For this study, the

variants predicted as possibly damaging and probably damaging were grouped together as deleterious variants.

4. **Xvar**. The web-based method Xvar version 0.75 beta was used with all default settings. The variants were submitted along with their Uniprot accession IDs. They were predicted as neutral, low, medium and high. For this study, the variants predicted as neutral and low were grouped together as neutral variants and the variants predicted as medium and high were grouped together as deleterious variants.

All field tests using the four algorithms were run during July 23-27, 2010 on a Mac OS X 10.5.8. Safari 5.0 was used to access the web-based methods.

3.1.2 Locus-Specific Databases

Sets of missense mutations with known functionality are needed to compare the algorithms. Locus-specific databases (LSDBs), which are curated collections of sequence variants in genes associated with disease [Greenblatt et al., 2008], can be used as a gold standard containing both neutral and deleterious variants. The variants from the LSDBs are evaluated by the algorithms which allow a comparison of the algorithms for sensitivity, specificity and receiver operating curves. Tavtigian et al. (2008) argue the best data sets for comparing these algorithms are LSDBs which are curated by individuals or groups specialized in the analysis of each specific gene [Chan et al., 2007, Chao et al., 2008]. A description of LSDBs for cancer susceptibility genes available on the internet can be found in Greenblatt et al. (2008). We evaluated the algorithms on four sets of variants with known functionality from BRCA1 (OMIM: 113705), MSH2 (OMIM: 609309), MLH1 (OMIM: 120436) and TP53

(OMIM: 191170) cancer associated genes.

1. The online Breast Cancer Information Core (BIC) [Szabo et al., 2000] mutation database (http://research.nhgri.nih.gov/bic/) was used to identify neutral ($n = 16$) and deleterious ($n = 17$) mutations from BRCA1. The steering committee of the BIC has manually reviewed available data from the literature and likelihood ratios [Goldgar et al., 2004] to define clinically relevant or benign missense changes using three categorizations of clinically relevant (yes, no or unknown).

2. The MLH1 missense mutations were obtained from two papers [Chan et al., 2007, Chao et al., 2008]. The first paper performed mismatch repair functional assays to identify neutral ($n = 10$) mutations with wild-type activity and deleterious ($n = 18$) mutations with impaired mismatch repair activity. These MLH1 mutations were compared in a previous paper [Chan et al., 2007] that used the three algorithms SIFT, Align-GVGD and PolyPhen, but they did not compare the results using different alignments. The second paper compiled a list of all known MLH1 missense mutations from several LSDBs with supporting data which was used as rigorous criteria to classify the variants as neutral ($n = 18$) or deleterious ($n = 37$). Subtracting the overlap between the papers yielded a total of ($n = 21$) neutral and ($n = 39$) deleterious variants.

3. The MSH2 variants were obtained from the same two papers as the MLH1 variants above [Chan et al., 2007, Chao et al., 2008]. The first paper identified neutral ($n = 3$) mutations with wild-type activity and deleterious ($n = 11$) mutations with impaired mismatch repair activity. These MSH2 mutations were also compared in the paper. The second paper compiled classified neutral ($n = 8$) or deleterious ($n = 13$) variants with the same criteria as above. Subtracting

the overlap between the papers yielded a total of ($n = 11$) neutral and ($n = 19$) deleterious variants.

4. The online TP53 database from the International Agency for Research on Cancer (IARC) was used to identify neutral ($n = 4$) and deleterious ($n = 140$) mutations from the TP53 gene (http://www-p53.iarc.fr/). A description of inclusion criteria for the polymorphisms and germline deleterious variants is given [Olivier et al., 2002].

3.1.3 Comparing the Accuracy of *in silico* Methods

In this analysis we compare the predicted functionality of the same set of curated missense mutations in cancer-associated genes using existing algorithms SIFT, Align-GVGD, PolyPhen-2 and Xvar. In addition, we provided SIFT, Align-GVGD and Polyphen-2 the same four sequence alignments for each gene analyzed to determine the impact of the alignment on prediction. Xvar is excluded from this latter analysis because it currently does not accept multiple sequence alignments as input.

Several statistical measures of performance were used to compare the performance of the algorithms for each of the four sets of mutations with known functionality. Using the notation of true positives (TP), true negatives (TN) false positives (FP) and false negatives (FN), we compute sensitivity as TP / (TP + FN) (probability of identifying true deleterious mutations) and specificity as TN / (TN + FP) (probability of identifying true neutral mutations). Some algorithms provide more than two prediction categories, e.g. neutral, possibly and probably damaging for Polyphen-2. Therefore, as described above for each method, we grouped the output into two categories deleterious and neutral based on similar groupings in previous studies [Chan et al., 2007].

A receiver operating characteristic (ROC) curve [Fawcett, 2006] is a technique that allows combining the mutation data and visualizing the performance of the algorithm with the native alignment and the three additional alignments provided (treated as a probabilistic classifier in this case). The ROC graph is a two dimensional graph that plots sensitivity against 1 - specificity depicting the relative tradeoffs between the true positives and false positives. ROC curves can be based on discrete or continuous classifiers. An algorithm that only reports a finite set of prediction categories, such as neutral or deleterious is a discrete classifier. However, if the algorithm reports a continuous score, being the degree or probability of a mutation belonging to a prediction category then it is a continuous classifier. We have reported the ROC curves in Figure 3.3 using the associated continuous scores available for any mutation tested in each algorithm (in fact, these scores underlie the discrete classifications). Accuracy is measured by the area under the ROC curve (AUC); an area of 1 corresponds to a perfect prediction, whereas an area of 0.5 corresponds to a pure chance prediction. AUC less than 0.5 may be interpreted as a systematically incorrect prediction. The AUC of a given classifier can be represented as the probability that given an alignment the algorithm will rank a randomly chosen deleterious mutation higher than a randomly chosen neutral mutation [Fawcett, 2006]. In our case, we use an AUC formula equivalent to the Wilcoxon test of ranks [Hanczar et al., 2010]. The confidence intervals of the estimated AUC values are identical with the confidence intervals of the Wilcoxon rank statistic [Hogg and Tanis, 2006]. The ROC curves and AUC values for all algorithm/alignment pairs were computed using the ROCR package in R [Sing et al., 2005].

BRCA1 Tumor Suppressor Gene

The native BRCA1 sequence alignments were built for the four algorithms as described in Methods. In addition, the three non-native alignments were used as inputs in each of the three algorithms, SIFT, Align-GVGD and PolyPhen-2 (see Table 3.1 for the number of sequences in each alignment). A description of the set of the well-characterized neutral ($n = 16$) and deleterious ($n = 17$) BRCA1 variants from the BRCA1 LSDB is described in Methods.

Figure 3.1A shows the output of each algorithm using the neutral ($n = 16$) BRCA1 variants when given the same four alignments. We found the algorithm PolyPhen-2 to be the least sensitive to the varying alignments. The Align-GVGD algorithm was the most sensitive to the varying alignments because the algorithm will predict all variants neutral, regardless of pathogenicity, when provided an alignment with a large number of sequences such as the PolyPhen-2 and Uniprot 50% alignments. Although this translates to an apparently high specificity for the Align-GVGD algorithm, this feature of the algorithm leads to the prediction of neutrality being solely based on the number of sequences in the alignment. Surprisingly, we see that algorithms do not necessarily perform best using their own alignment (Table 3.1). For example, the SIFT algorithm has the highest specificity using the Align-GVGD alignment (compared to its own) possibly because the Align-GVGD alignment is only made up of orthologs [Ng and Henikoff, 2002]. We also found that the SIFT algorithm overcalls neutral variants as deleterious, low specificity, as previously noted by others [Mathe et al., 2006, Karchin et al., 2008]. Using its own or native alignment, the Xvar algorithm has specificity (Table 3.2) that is equal to or smaller than the specificities of the other three algorithms using their optimal alignment (Table 3.1).

The results in Figure 3.1B show the output of each algorithm for the deleterious

Table 3.1 : Specificity and Sensitivity Summary for All Four Genes BRCA1 ($n = 16$ Neutral, $n = 17$ Deleterious), MSH2 ($n = 11$ Neutral, $n = 19$ Deleterious), MLH1 ($n = 21$ Neutral, $n = 39$ Deleterious), and TP53 ($n = 4$ Neutral, $n = 140$ Deleterious) using the Same Four Alignments

			Algorithms					
			SIFT		Align-GVGD		PolyPhen-2	
Genes	Alignments	No. sequences in alignment	Spec (%)	Sens (%)	Spec (%)	Sens (%)	Spec (%)	Sens (%)
BRCA1	SIFT	360	31.3	94.1	93.8	64.7	56.3	94.1
	Align-GVGD	13	75.0	70.6	93.8	70.6	31.3	82.4
	PolyPhen-2	279	43.8	47.1	100.0	0.0	37.5	76.5
	Uniprot 50%	275	50.0	47.1	100.0	0.0	25.0	76.5
MSH2	SIFT	45	45.5	89.5	81.8	21.1	27.3	94.7
	Align-GVGD	14	18.2	89.5	54.5	89.5	27.3	89.5
	PolyPhen-2	56	45.5	94.7	100.0	0.0	36.4	89.5
	Uniprot 50%	53	18.2	94.7	100.0	0.0	18.2	94.7
MLH1	SIFT	29	52.4	71.8	81.0	48.7	42.9	66.7
	Align-GVGD	11	57.1	97.4	52.4	97.4	42.9	97.4
	PolyPhen-2	191	61.9	89.7	100.0	0.0	66.7	89.7
	Uniprot 50%	45	52.4	82.1	100.0	0.0	42.9	97.4
TP53	SIFT	72	75.0	84.3	100.0	57.9	75.0	87.1
	Align-GVGD	9	75.0	85.7	100.0	82.1	25.0	92.1
	PolyPhen-2	93	100.0	73.6	100.0	0.0	100.0	84.3
	Uniprot 50%	113	100.0	32.1	100.0	0.0	75.0	90.0

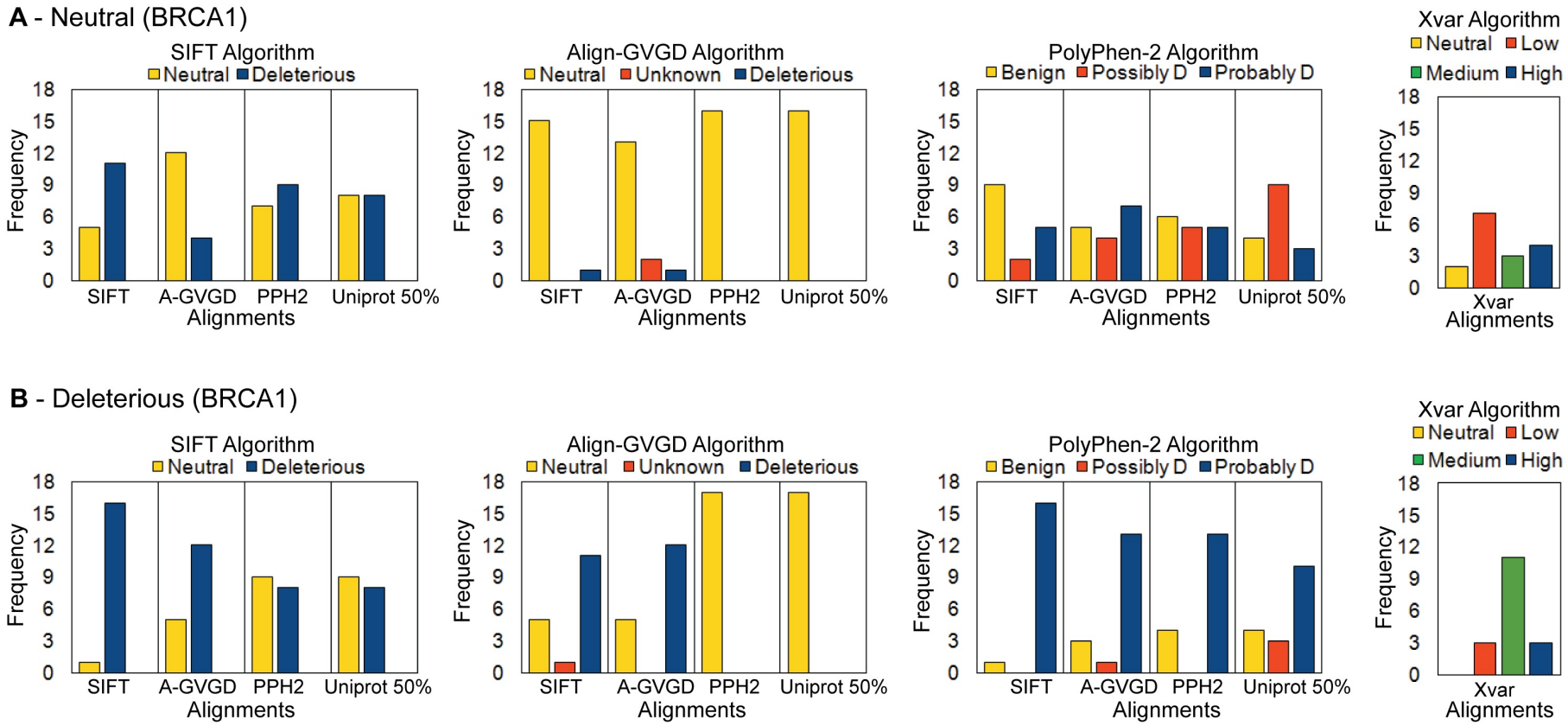

Figure 3.1 : **(A)** Predictions of neutral (n=16) BRCA1 missense mutations using three algorithms with four alignments each. The four alignments are represented by SIFT (SIFT), Align-GVGD (A-GVGD), PolyPhen-2 (PPH2), and Uniprot 50% (Uniprot 50%). The prediction categories for PolyPhen-2 Possibly Damaging and Probably Damaging have been abbreviated to Possibly D and Probably D. The algorithm Xvar employs its own alignment. **(B)**: Predictions of deleterious (n=17) BRCA1 missense mutations using three algorithms with four alignments each.

Table 3.2 : Specificity and Sensitivity Summary Using the Xvar Default Alignment for the Four Genes BRCA1, MSH2, MLH1, and TP53

Genes	Specificity (%)	Sensitivity (%)
BRCA1	56.3	82.4
MSH2	27.3	100
MLH1	33.3	100
TP53	50.0	95.7

(n = 17) BRCA1 variants when given the same four alignments. The Align-GVGD algorithm shows a poor sensitivity, again because it incorrectly predicts all 17 deleterious variants as neutral when provided large number of sequences, as it is the case for PolyPhen-2 and Uniprot 50% alignments. The algorithm PolyPhen-2 has the highest sensitivity using the SIFT alignment which is another example of an algorithm performing best with an alignment other than its own (Table 3.1). When comparing the Xvar algorithm using its native alignment to the other algorithms, we see Xvar has a high sensitivity (Table 3.2) that is similar to the sensitivities reported from the other algorithms using their optimal alignment (Table 3.1).

MSH2, MLH1 Mismatch Repair Genes and TP53 Tumor Suppressor Gene

The native MSH2, MLH1 and TP53 sequence alignments were built for the four algorithms as described in Methods. In addition, the three non-native alignments for each gene were used as inputs in each of the three algorithms, SIFT, Align-GVGD and PolyPhen-2 (see Table 3.1 for the number of sequences in each alignment). The three sets of variants from MSH2 (n = 11 neutral, n = 19 deleterious), MLH1 (n = 21 neutral, n = 39 deleterious) and TP53 (n = 4 neutral, n = 140 deleterious) are described in Methods. Overall we found similar results to BRCA1 for variants

from these three cancer genes (Table 3.1). However, SIFT algorithm reports higher sensitivities using the MSH2 and MLH1 variants compared to the BRCA1 variants. We also note that although for BRCA1 the highest specificity of the SIFT algorithm was seen by using the Align-GVGD alignment this was not true for the other three genes. The results for MSH2, MLH1 and TP53 using the Xvar algorithm are given in Table 3.2 which again demonstrates the algorithm using its native alignment reports similar specificity values to the other algorithms. When comparing sensitivity, Xvar using its native alignment reports higher sensitivities (Table 3.2) that is equal to or greater than the sensitivities of the other algorithms using their best alignment (Table 3.1) for all three genes.

Overall Sensitivity and Specificity

A boxplot summary of the sensitivity and specificity values for the three algorithms, which combines the gene- and alignment-specific information, illustrates how much variation is caused by employing different alignments (Figure 3.2).

For each alignment, the four sensitivity values are computed by grouping the mutations within each of the four genes BRCA1, MSH2, MLH1 and TP53, yielding a total of 16 sensitivity values for each algorithm. We note that there are only four neutral TP53 variants which may inflate the specificity values; therefore we excluded TP53 specificity values in the figure and used only 12 specificity values for each algorithm. We also computed confidence intervals for the sensitivity and specificity estimates, using the Wilson score method [Agresti, 2002]. The results from Figure 3.2 show PolyPhen-2 and SIFT both have a high median sensitivity of 0.90 and 0.85, respectively. We note PolyPhen-2 and SIFT both have similar median specificity values, 0.40 and 0.52, respectively, highlighting that the specificities are significantly lower

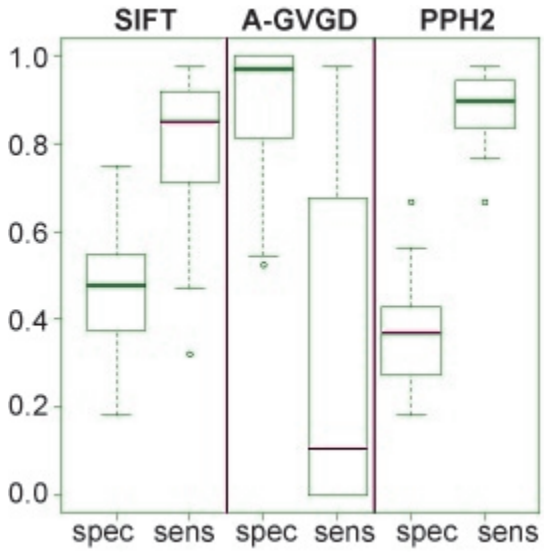

Figure 3.2 : Boxplots of specificity (spec) and sensitivity (sens) for each algorithm as given in Table 1. Sensitivity values are reported using all four genes BRCA1, MSH2, MLH1, and TP53, but TP53 is excluded in specificity values to account for potential bias given that there are only four neutral variants. The three algorithms are represented by SIFT (SIFT), Align-GVGD (A-GVGD), and PolyPhen-2 (PPH2).

than the sensitivities. Thus, these algorithms are more likely to make mistakes by calling neutral variants deleterious. For both sensitivity and specificity PolyPhen-2 has a smaller interquartile range (IQR) than SIFT, which means that PolyPhen-2 is less sensitive to the sequence alignment employed. As noted previously, Align-GVGD is very sensitive to the algorithm employed. The high specificity seen in Align-GVGD is misleading because the algorithm predicts all variants neutral, regardless of pathogenicity, when using alignments with a large number of sequences. This feature of the algorithm also results in low sensitivity with large IQR. Thus, even though Align-GVGD performs well using its own alignment, it is more dependent on the alignment employed than either SIFT or PolyPhen-2. The results of this analysis are that only PolyPhen-2 and SIFT are appropriate for use with non-native

alignments that are not manually curated, with PolyPhen-2 modestly outperforming SIFT. The corresponding boxplot for the Xvar algorithm is not provided because Xvar requires the use of its native alignment; however, the median sensitivity is 0.98 and the median specificity (excluding TP53) is 0.33.

Receiver Operating Characteristic curves

Each algorithm provides a quantitative probability or score as output as well as a prediction category, e.g. probably damaging for Polyphen-2. This enabled us to compare alignment-specific information for each algorithm using the concept of receiver operating characteristic (ROC) curves to provide a succinct graphical summary of all four algorithms, treated as continuous classifiers (Figure 3.3A and Figure 3.3B; also, see Statistics Section in Materials and Methods). Align-GVGD and PolyPhen-2 algorithms performed best using their native alignment, but the SIFT algorithm had a higher AUC when using an alignment, manually curated Align-GVGD, other than its own. When employing the optimal alignment for each algorithm, the AUC values are 79% for all four algorithms. The PolyPhen-2 algorithm is shown to be the least dependent on the alignment employed as seen by the nearly overlapping ROC curves and similar AUC values for all four alignments provided. In comparison the SIFT and Align-GVGD algorithms show much greater variation in their ROC curves employing different alignments. As seen in Figure 3.2, Figure 3.3 shows PolyPhen-2 and SIFT are the only two methods appropriate for use with non-native algorithm-generated alignments. The area under the ROC curve (AUC) values are reported in Table 3.3 with the associated confidence intervals. When we rank the averaged AUC values for each alignment strategy we obtain 0.790, 0.766, 0.674, and 0.579 for the Align-GVGD, SIFT, PolyPhen-2 and Uniprot alignments, respectively.

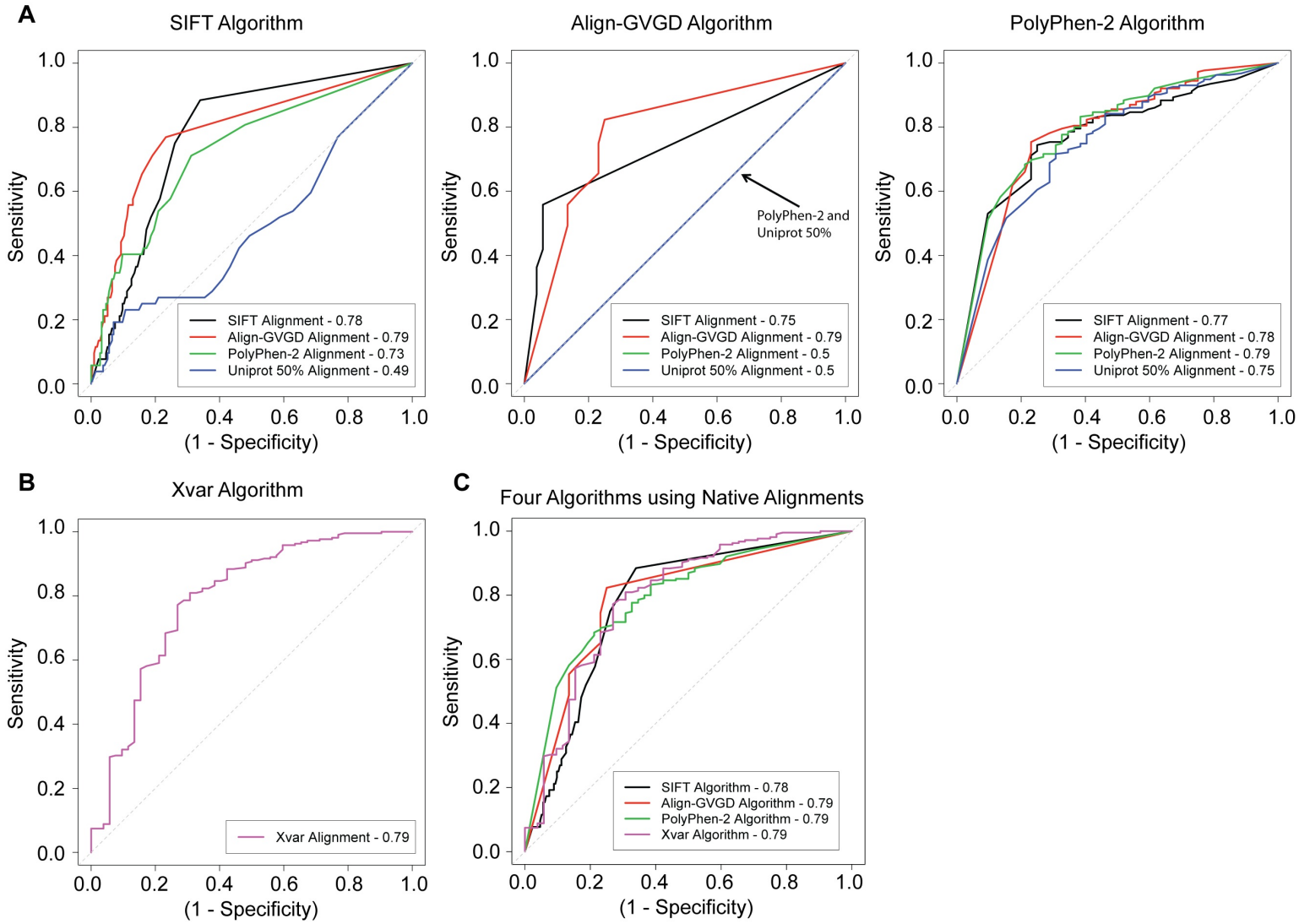

Figure 3.3 : **(A)** Receiver operating characteristic (ROC) curves using probabilities and scores associated with each prediction for each of the three algorithms SIFT, Align-GVGD, and PolyPhen-2. For each algorithm, four colored lines (black, red, green-blue) are drawn representing the four alignments used in each algorithm. The area under the curve (AUC) is reported in the legend. **(B)** Receiver operating characteristic (ROC) curve using the four genes BRCA1, MSH2, MLH1, and TP53 from the Xvar algorithm. The pink line drawn represents the Xvar alignment. **(C)** ROC curves comparing the performance of the four algorithms using their own native alignments.

Table 3.3 : Area Under the Curve (AUC) from Receiver Operating Curves for Each Algorithm Using Each Alignment Using Probabilities and Scores Associated with Each Mutation Prediction

Algorithms	Alignments	AUC	CIs
SIFT	SIFT	0.777	(0.690, 0.865)
	Align-GVGD	0.790	(0.702, 0.877)
	PolyPhen-2	0.730	(0.643, 0.818)
	Uniprot 50%	0.487	(0.400, 0.575)
Align-GVGD	SIFT	0.747	(0.659, 0.835)
	Align-GVGD	0.791	(0.703, 0.878)
	PolyPhen-2	0.500	(0.412, 0.588)
	Uniprot 50%	0.500	(0.412, 0.588)
PolyPhen-2	SIFT	0.773	(0.686, 0.861)
	Align-GVGD	0.779	(0.692, 0.867)
	PolyPhen-2	0.792	(0.704, 0.879)
	Uniprot 50%	0.750	(0.662, 0.838)
Xvar	Xvar	0.790	(0.703, 0.878)

Given that most investigators will utilize online tools where the algorithm employs its native alignment we compared the ROC curves for the four algorithms using their native alignments (Figure 3.3C). This analysis demonstrates no significant differences in the shape of the curve or AUC values (between 78-79%) for all four algorithms. The same analysis utilizing the optimal alignment for each algorithm results in only a small difference in the AUC for the SIFT algorithm increasing from 78 to 79%.

3.2 Disagreement Among Predictions of Functionality

Chan et al. (2007) compared four methods: SIFT, Align-GVGD, PolyPhen and the BLOSUM62 matrix, using the native alignments supplied by the program or manually curated for Align-GVGD. In the paper each method individually had a limited overall predictive value (72.9-82.0%), but when all four methods agree (62.7%), the overall

predictive value increased to 88.1%. Karchin et al. (2009) argued these algorithms have major similarities underneath the lid of each method and the correlation of their outputs is the result of similarity of their inputs, which is not a cause for increased confidence. Chun and Fay (2009) suggested differences between missense predictions from the algorithms may be due to differences in the sequences and/or alignments used to identify evolutionary conserved mutations. We directly tested this idea by comparing the predictions of the SIFT, Align-GVGD and PolyPhen-2 algorithms by supplying the same four alignments to each algorithm. Surprisingly we found a given algorithm did not necessarily perform best using the alignment provided by the creator of the algorithm. For example, the PolyPhen-2 algorithm reported higher sensitivities in all four genes using alignments other than its own and SIFT had a slightly higher AUC when provided the Align-GVGD alignment containing only orthologs as originally predicted by Ng and Henikoff (2002). The three algorithms SIFT, PolyPhen-2 and Xvar all had a high sensitivity, but low specificity implying these algorithms may overcall neutral variants deleterious. This feature was most pronounced for Xvar with higher sensitivities and lower specificities than most of the other algorithms. We showed Align-GVGD was the most affected by alignment employed, performing well when using manually curated alignments, but calling all variants neutral when alignments contain a large number of sequences. Thus, for large-scale sequencing experiments Align-GVGD would require development of alignments with orthologous sequences through evolution for all genes. Conversely, the PolyPhen-2 algorithm was shown to be the least sensitive to alignment provided with nearly overlapping ROC curves. The ROC analysis resulted in AUC values of 78-79% for all four algorithms using native alignments and 79% when using the optimal alignment for each algorithm which shows despite the differences in predictions from

the algorithms and alignments the overall performance of these four commonly used methods is similar.

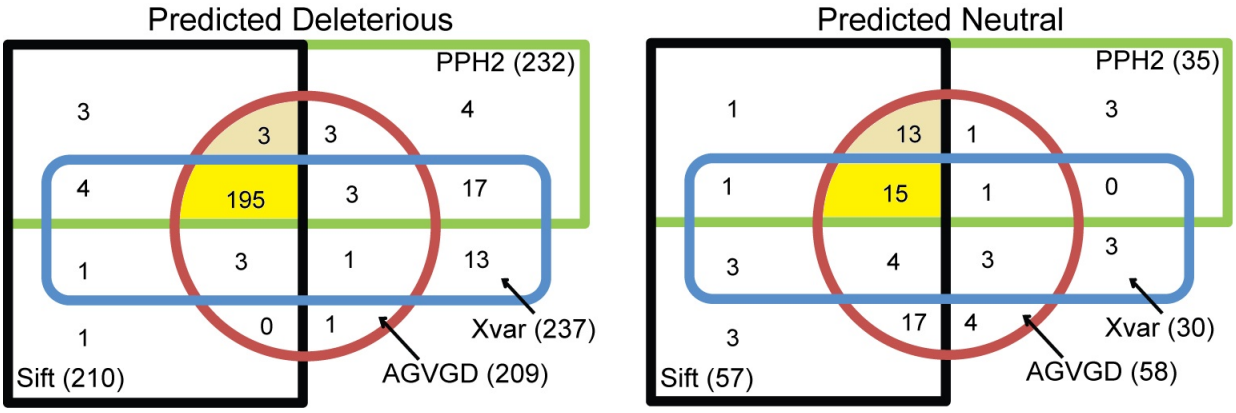

Figure 3.4 : Predictions of neutral and deleterious mutations with the SIFT, Align-GVGD, and PolyPhen-2 algorithms using the Align-GVGD alignment and the Xvar algorithm using its own alignment. We also depict the exclusive overlap of the predictions between the four algorithms to show their agreement (dark yellow) and between the three algorithms SIFT, Align-GVGD, and PolyPhen-2 (light yellow).

Karchin et al. (2009) further argued that when the outputs from the algorithms SIFT and PolyPhen differ, it is more likely due to using different protein sequence alignments compared to the differences in scores used to classify the variants. From our experimental design, we were able to directly test this hypothesis. Using a Venn diagram we depict the disjoint classification of variants predicted deleterious and neutral in Figure 3.4, respectively by different algorithms all employing the Align-GVGD alignment (as all three algorithms performed well with this alignment). When considering the predicted deleterious mutations the four algorithms agree on 195 mutations (77%) using the Align-GVGD alignment, but the SIFT, Align-GVGD and PolyPhen-2 algorithms agree on an additional three mutations for a total of 198 mutations (79%). Of this 195 only 181 are actually classified as deleterious by the LSDB. Interestingly,

Chun and Fay (2009) compared the predicted deleterious mutations from the three algorithms SIFT, PolyPhen and Likelihood Ratio Test (LRT) resulting in a very low overlap of 5%, but when we perform a similar analysis employing the algorithms' own alignment, we see a much higher overlap of 70%. When considering the predicted neutral mutations, the four algorithms only agree on 15 mutations (20%); excluding Xvar results in agreement for another 13 mutations for a total of 28 mutations (39%) which again demonstrates problems in predicting variants to be neutral. Only 11 of the 15 variants are classified as neutral by the LSDB implying even when provided the same alignment the algorithms make different predictions. Further research is needed to understand the underlying differences in these algorithms. Thus, in order to predict missense mutation functionality, the researcher should consider optimizing both the algorithm and sequence alignment employed.

3.3 Discussion

As we have shown in this chapter, predicting the impact of missense mutations on protein function depends on the algorithm used, the type of sequence alignment provided, and on the number of sequences in the alignment leading to multiple interpretations for each mutation. In addition to problems of interpretation there are technical difficulties as well. In our experience, when simply submitting a list of missense mutations to an algorithm the user must be able to: (1) manipulate the input format specified by each algorithm, (2) build an optimal protein sequence alignment, if required, (3) be knowledgeable of Unix system commands, (4) interpret server error messages, and (5) transform the output to a working format for further studies. Standard input and output formats are needed to alleviate the burden on the user. Also, tools to create informative protein sequence alignments for each protein are necessary to accurately

predict the impact of missense mutations on protein function. An additional source of error in prediction when analyzing sequence variants identified through disease status is that all algorithms focus on the missense change encoded by the variant when in reality the sequence variant may also impact gene expression for example through alternative splicing of the messenger RNA.

Finally, great caution should be taken when comparing the accuracy of new *in silico* methods. A recent article [Acharya and Nagarajaram, 2012] was published describing a new method called Hansa, which classifies missense mutations into neutral and deleterious categories. However, the authors did not provide sufficient details about their algorithm, which resulted in a number of concerns about the appropriateness and application of statistical methods that compare Hansa with existing algorithms. The authors stated their method outperformed other known methods such as PolyPhen-2 and SIFT by comparing the ROCs of Hansa to the ROCs of the other algorithms. In their Table 2, a direct comparison of the ROCs was made by employing a benchmark dataset called HumVar originally described in Capriotti et al. (2006) and employed in Adzhubei et al. (2010), which compares the true positive rates between algorithms for a fixed false positive rate. As shown in this chapter, ROCs require a probability or continuous score associated with each prediction to compute TPRs and FPRs as the discrimination threshold is varied [Pepe, 2004]. For example, the ROC of PolyPhen-2 was based on the naive Bayes probability provided by the algorithm itself and the ROC of SIFT was based on the SIFT score [Adzhubei et al., 2010]. As described in the publication, Hansa is based on support vector machine (SVM) method, which uses a set of 10 discriminatory features to classify missense mutations as neutral or deleterious. SVMs are nonprobabilistic classifiers [Hastie et al., 2009], and consistently, there is no probability or continuous score associated with each

prediction, and thus, an ROC analysis does not seem obviously feasible for this algorithm. In the publication, there is no mention of what continuous score or probability was used to calculate the TPRs of Hansa for a fixed FPR. Therefore, it is unclear how they might attain various TPRs for a given FPR because there is no varying threshold defined.

We compared Hansa with other algorithms using the independent data set of $n = 267$ mutations from cancer-associated genes [Hicks et al., 2011], which Acharya and Nagarajaram (2012) use as a validation data set to the HumVar data that Hansa was trained on. We originally used this well-characterized data to compare the TPRs and FPRs of several algorithms using their native protein sequence alignments and to evaluate the impact of the predictions when the algorithms were supplied other alignments. Because Hansa does not provide a probability or continuous score associated with each prediction, we could not provide the ROC curves and could only calculate the TPRs and FPRs for each algorithm. Hansa seems to perform comparably to the other algorithms (Figure 3.5).

In addition, as a way to compare the improvement of TPR in Hansa over the other algorithms, the authors inappropriately performed a paired t-test. Because they are comparing proportions, it would be preferable to use for example a test for a difference in proportions with a correction for multiple testing. Furthermore, to measure the performance of the SVM, the authors state they use a n-fold cross-validation and leave-one-out cross-validation (LOOCV) to assess the generalization and stability of their method. Unfortunately, they do not report the parameter estimates of the SVM and do not report the n-fold cross-validation error. They only report a LOOCV error, which makes it difficult to assess the validity of this analysis. In summary, a thorough statistical assessment is needed when comparing these *in silico* algorithms.

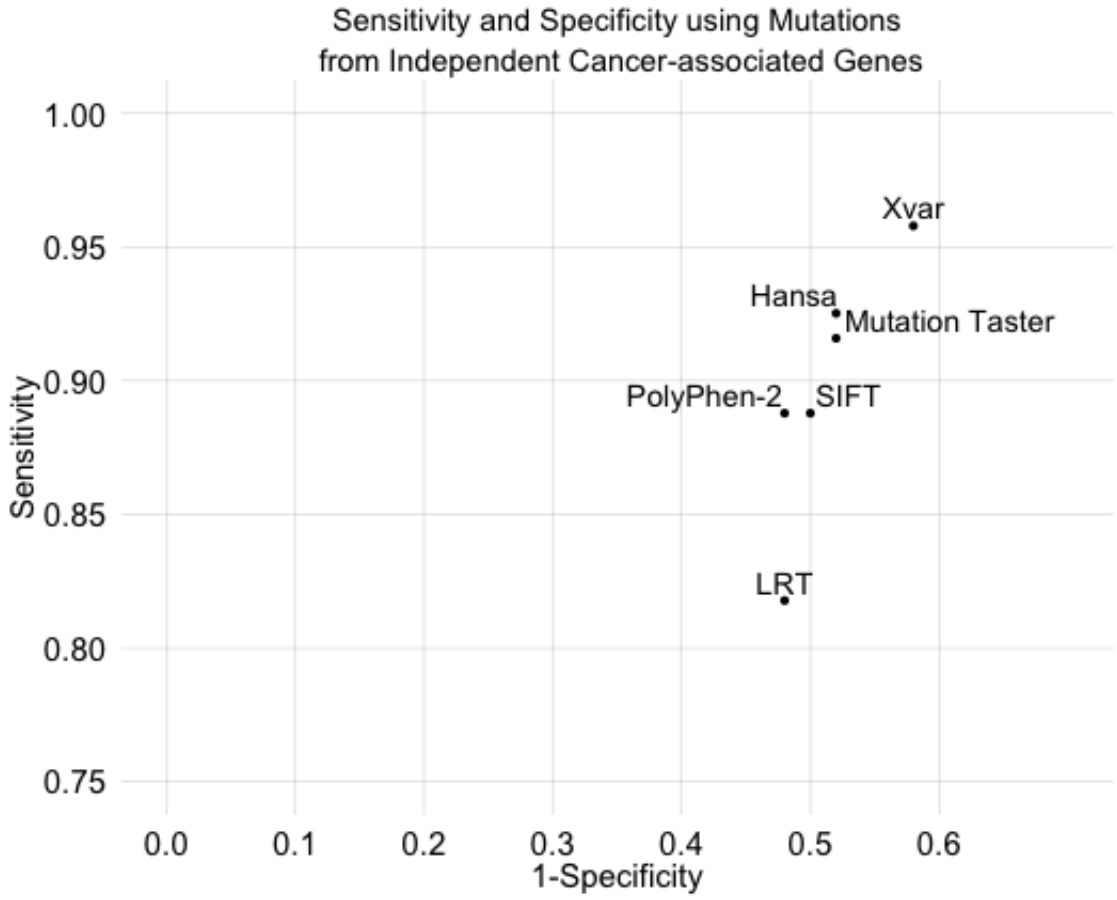

Figure 3.5 : Predictions from the following four algorithms were taken from dbNSFP [Liu et al., 2011]: Likelihood Ratio Test (LRT), Mutation Taster, PolyPhen-2, and SIFT; predictions from the last two algorithms were produced on their respective web pages: Hansa [Acharya and Nagarajaram, 2012] and Xvar [Reva et al., 2011].

Chapter 4

A New Method for Interpreting the Functionality of Missense Mutations

Thousands of missense mutations with unknown biological significance are reported by genome-scale sequencing projects. As shown in Chapter 3, many computational or *in silico* methods have been developed to predict the functionality of missense mutations, but surprisingly there is a high degree of disagreement among the predictions produced by these methods even though the majority of these methods base their predictions on similar information (the use of evolutionary conservation as a measure of pathogenicity). These discordant functional predictions often leave researchers without guidance in how to prioritize the mutations identified for further evaluation in biological functional assays. In this chapter, we develop two statistical models based on the capture-recapture paradigm which combine the discordant functional predictions in a statistically rigorous manner and estimate a unified posterior probability of functionality or pathogenicity for each missense mutation. Unlike previous methods, our probabilistic approach requires no training set or calibration and estimates the accuracy (sensitivity and specificity) of each individual *in silico* method in the absence of a gold standard by taking advantage of the fact these methods disagree. In Section 4.2, we develop two models referred to as postMUT and postMUT (simple) and derive the parameter estimates for the Expectation-Maximization algorithm. In Section 4.3, we give several applications such as we show our estimates of sensitivity and specificity of the *in silico* algorithms (without employing a gold standard) match

the estimates of sensitivity and specificity when a gold standard is available. The posterior probability of pathogenicity introduced in this chapter is a statistical tool scalable to the exome which may be used to infer the functionality of missense mutations and can be easily incorporated in downstream analyses such as disease gene prioritization tools ultimately inferring candidate genes.

4.1 Combining Discordant Predictions of Missense Mutation Functionality using Capture-Recapture Methods

Determining the consequences of genetic variation is a major challenge in bioinformatics and genomics. It is estimated each individual genome differs from a reference genome at 3.0-3.5 million variants whereas each individual exome differs at 20,000-30,000 variants of which 10,000 are predicted to be nonsynonymous changes, splice site changes or indels [Robinson et al., 2011, Gonzaga-Jauregui et al., 2012]. Interpreting the functional consequences of these nonsynonymous changes is an important step in identifying and determining the clinical importance of disease susceptibility mutations. In particular, the interpretation of missense mutations (point mutation in which a single nucleotide is changed resulting in amino acid change) has remained a difficult task because missense mutations do not necessarily impact protein function.

Many computational or *in silico* algorithms have been developed to predict the impact of missense mutations on protein function. Several reviews of these methods are available [Mooney, 2005, Ng and Henikoff, 2006, Karchin, 2009, Thusberg and Vihinen, 2009, Jordan et al., 2010, Thusberg et al., 2011]. In general, these methods can be classified into three groups: first-principles methods, trained classifiers or genomic annotation tools [Cooper and Shendure, 2011]. The majority of first-principles methods and

trained classifiers use protein sequence alignments as input because they are based on a similar idea of using evolutionary conservation or a combination of phylogenetic information with protein structure or other sequence annotations as a measure of the pathogenic effect of missense mutations on protein function [Jordan et al., 2010]. We have previously shown predictions from these *in silico* methods are highly dependent on the input parameters used in each algorithm (such as phylogenic scope and quality of the protein sequence alignments) and make different predictions even when provided the same protein sequence alignment [Hicks et al., 2011].

A major problem with these *in silico* methods is there is a high degree of disagreement among the functional predictions even though these methods rely on similar evolutionary information to assess functionality. Several studies have investigated the agreement of predicted deleterious missense mutations between these methods and report varying estimates ranging from 3.5% to 77% [Chun and Fay, 2009, Jaffe et al., 2011, Hicks et al., 2011, Gray et al., 2012]. This suggests the functional predictions produced from these *in silico* methods may be context-dependent (missense mutations play different roles in different diseases and therefore their functional effect may vary) and mutation set-dependent (certain methods may be calibrated to perform more accurately on specific sets of mutations).

Some previously proposed solutions referred to as umbrella methods [Sunyaev, 2012] or consensus tools such as Condel [González-Pérez and López-Bigas, 2011] or Carol [Lopes et al., 2012] have been developed to combine the functional predictions from these algorithms based on some weighted average of scores from the individual *in silico* methods. The problem with these approaches is they do not account for the sensitivity (probability of calling the variant deleterious if it is deleterious) or specificity (probability of calling the variant neutral if it is neutral) of each method and do not

seem to be based on rigorous statistical principles. Others have combined functional predictions using logistic regression [Thompson et al., 2013], but this requires a gold standard or set of mutations with known functionality to estimate the coefficients in the regression model. This approach is not immediately scalable to the exome because it requires individual calibration for each gene and requires the use of locus-specific databases (LSDBs) or manually curated collections of sequence variants associated with diseases. Because validated exome-scale LSDBs are not available, most groups use sets of mutations only weakly associated with disease, but this leads to possible biases in comparisons between *in silico* methods.

In this chapter, we take a maximum likelihood based approach [Lehmann and Casella, 1998, Shao, 2003] to combine the discordant functional predictions in a statistically rigorous manner and to estimate a unified posterior probability of functionality or pathogenicity for each missense mutation. Unlike previous methods, our approach, referred to as postMUT and postMUT (simple), requires no training set or calibration and estimates the sensitivity and specificity of each individual *in silico* method in the absence of a gold standard by taking advantage of the fact these methods disagree. The idea is borrowed from the technique named capture-recapture [Otis et al., 1978, Pollock, 1982] which originated in ecology as a way to estimate population sizes. When gold standards or sets of mutations with known functionality are available, researchers may use these to fine-tune parameters such as evolutionary depth of the protein sequence alignments as input to the prediction algorithms and to optimize performance of the algorithm. To test the validity of this approach we show our estimates of sensitivity and specificity of the *in silico* methods (without employing a gold standard) match the estimates of sensitivity and specificity when a gold standard is available. These posterior probabilities of pathogenicity for missense mutations introduced in this chapter

may be used to further prioritize the mutations identified for evaluation in biological functional assays and can be incorporated in downstream analyses of exome-scale datasets such as disease gene prioritization tools ultimately inferring candidate genes.

4.2 Bernoulli Mixture Models: A Maximum Likelihood Approach

The postMUT and postMUT (simple) models are defined as mixtures of Bernoulli probability distributions with the weight parameter representing the overall proportion of deleterious mutations in a given set. The main principle of the postMUT (simple) model is that a given missense mutation can be defined as having a functional or neutral effect on protein function. These two exclusive possibilities are correspondingly coded as $Y = 1$ and $Y = 0$. Y is a latent (unobservable) variable. Given Y, each jth *in silico* method calls the variant functional or neutral (coded as $X_j = 1$ and $X_j = 0$) with different Y-dependent probabilities. These outcomes are observable. Somewhat surprisingly, these observations are frequently sufficient for estimation of the probabilities of $Y = 1$ and the probabilities of $X_j = 1$ and $X_j = 0$ given Y.

If we consider n *in silico* methods, then we may separate a set of mutations into disjoint classes or categories depending on which mutations are predicted deleterious (or neutral) by each method to depict the overlap or agreement between the methods. We previously provided an example of these disjoint classes (Hicks et al. 2011). Figure 4.1 gives two examples of Venn diagrams depicting the disjoint classes of mutations predicted deleterious by three commonly employed algorithms (SIFT, PolyPhen-2 and MutationAssessor) on two gold standard databases HumDiv and

HumVar. An important technical point is that we use terms functional and deleterious interchangeably to represent mutations that have a damaging effect on molecular function resulting in a negative effect on the fitness such as a loss of protein function in tumor suppressor genes or a gain of function in oncogenes [Sunyaev, 2012].

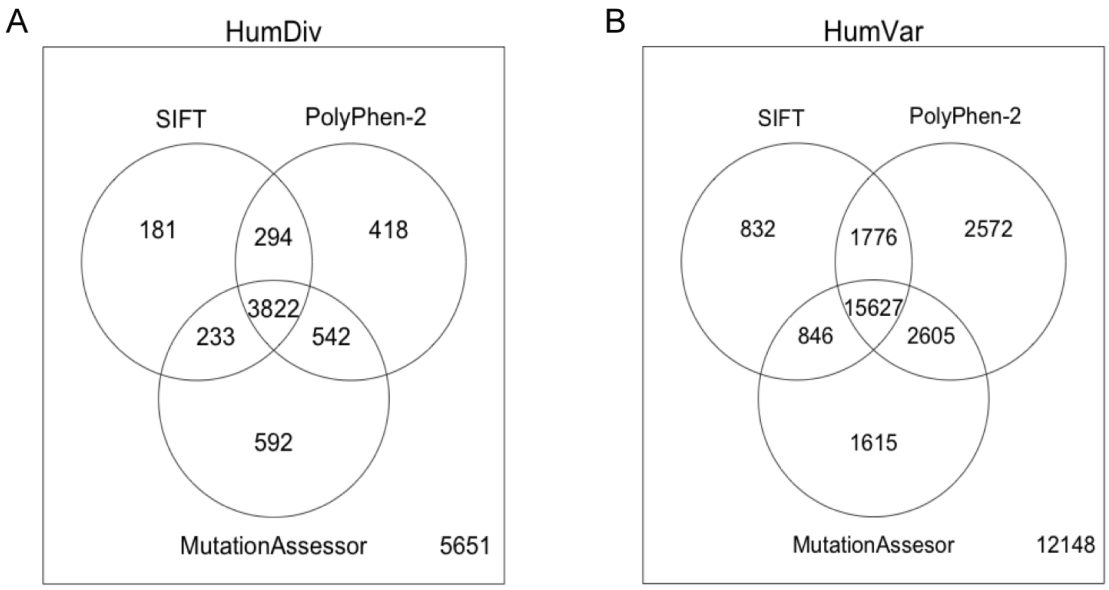

Figure 4.1 : Venn diagram of the number of mutations predicted deleterious from SIFT, MutationAssessor and PolyPhen-2 using: **(A)** mutations from HumDiv and **(B)** mutations from HumVar. The number outside the Venn diagrams represent the number of predicted neutral mutations by all three algorithms.

4.2.1 Formal Definition of Model

Formally, we define each missense mutation in a dataset (e.g. missense mutations reported from an individual exome) to have either a "deleterious" or a "neutral" effect on protein function. Consider a set of m mutations and n algorithms. Let $\mathbf{Y} = (Y_1, \ldots, Y_m)$ denote the "gold standard" or true functional status of the m

mutations where

$$Y_i = \begin{cases} 1 & \text{if } i\text{th mutation is truly deleterious} \\ 0 & \text{if } i\text{th mutation is truly neutral} \end{cases}$$

If a gold standard is available, we can simply calculate the sensitivity and specificity for each algorithm directly. Because gold standards are not widely available, Y_i is considered a latent variable or missing information. Instead, we only observe the predictions of functionality from the various *in silico* methods which we refer to as $\mathbf{X}$. Let $\mathbf{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_n)$ denote a set of predictions of missense mutation functionality from n methods where

$$X_{ij} = \begin{cases} 1 & \text{if } i\text{th mutation is predicted deleterious} \\ 0 & \text{if } i\text{th mutation is predicted neutral} \end{cases}$$

In silico methods often produce a continuous probability or score associated with each binary functional prediction. We do not use these continuous values, but we are interested in extending these models to incorporate them to improve the accuracy. To assess the degree of disagreement between the methods, disjoint categories of mutations predicted deleterious (or neutral) are calculated using X_{ij}. If there are n methods, then there are 2^n categories that the ith mutation could fall into. For example, Tables 4.1 and 4.2 are examples of how to label the 4 and 8 disjoint categories from $n = 2$ and $n = 3$ algorithms, respectively, which depend on the joint functional predictions X_i from the ith mutation.

Let $N(k)$ be the number of mutations in the kth category where

$$N(k) = \sum_{i=1}^{m} I_{[K(\mathbf{X}_i)=k]} \tag{4.1}$$

and $I_{[K(\mathbf{X}_i)=k]}$ is an indicator function for the kth category. We note the expectation $E(I_{[K(\mathbf{X}_i)=k]})$ is the probability of the ith mutation being in the kth category

Table 4.1 : Example labels for disjoint categories (1, ..., 2^n) which depend on the joint predictions $\mathbf{X}_i$ from using $n = 2$ *in silico* methods.

X_{i1}	X_{i2}	Category
0	0	$K(\mathbf{X}_i) = 1$
1	0	$K(\mathbf{X}_i) = 2$
0	1	$K(\mathbf{X}_i) = 3$
1	1	$K(\mathbf{X}_i) = 4$

Table 4.2 : Example labels for the 2^n disjoint categories which depend on the joint predictions $\mathbf{X}_i = (X_{i1}, X_{i2}, X_{i3})$ from using $n = 3$ algorithms.

X_{i1}	X_{i2}	X_{i3}	Category
0	0	0	$K(\mathbf{X}_i) = 1$
0	0	1	$K(\mathbf{X}_i) = 2$
0	1	0	$K(\mathbf{X}_i) = 3$
1	0	0	$K(\mathbf{X}_i) = 4$
0	1	1	$K(\mathbf{X}_i) = 5$
1	0	1	$K(\mathbf{X}_i) = 6$
1	1	0	$K(\mathbf{X}_i) = 7$
1	1	1	$K(\mathbf{X}_i) = 8$

$P(K(\mathbf{X}_i) = k)$ which uses the functional predictions corresponding to the kth category $\mathbf{X}_i = K^{-1}(k)$ (will be further discussed in Section 4.2.2).

Conditional on the true functional mutation status Y_i, we assume each jth method will predict the functionality of the ith mutation with a different sensitivity (probability of identifying true deleterious mutations) and specificity (probability of identifying

true neutral mutations). Therefore, we define the parameters

$$\begin{aligned} a_j &= P(X_{ij} = 1|Y_i = 0) \\ b_j &= P(X_{ij} = 1|Y_i = 1) \\ p &= P(Y_i = 1) \end{aligned}$$

where a_j can be interpreted as the probability of a false positive (or incorrectly predicting the mutation to be deleterious) for the jth method, b_j can be interpreted as the probability of a true positive (correctly predicting a deleterious mutation) for the j method and p is the probability of a deleterious mutation.

Next, we develop two statistical models referred to as postMUT (simple) and postMUT which are both mixtures of Bernoulli distributions. These models jointly estimate the sensitivity and specificity of each individual *in silico* method and the overall proportion of deleterious mutations.

4.2.2 Model Formulation

postMUT (simple) Model

The probability of observing the functional prediction for the ith mutation from the jth algorithm is

$$\begin{aligned} P(X_{ij} = x_{ij}) &= P(X_{ij} = x_{ij}|Y_i = 0)P(Y_i = 0) + P(X_{ij} = x_{ij}|Y_i = 1)P(Y_i = 1) \\ &= a_j^{x_{ij}}(1 - a_j)^{(1-x_{ij})}(1 - p) + b_j^{x_{ij}}(1 - b_j)^{(1-x_{ij})}p \end{aligned}$$

with unknown parameters $\theta = (a_1, \ldots, a_n, b_1, \ldots, b_n, p)$. If we assume conditional independence between n algorithms given knowledge of the true functionality for

each ith mutation, then the joint likelihood of $\mathbf{X}_i$ can be written as

$$\begin{aligned} P_{\mathbf{X}_i} &= P_{[\mathbf{X}_i|Y_i=0]}P_{[Y_i=0]} + P_{[\mathbf{X}_i|Y_i=1]}P_{[Y_i=1]} \\ &= \prod_{j=1}^{n} P_{[X_{ij}|Y_i=0]}P_{[Y_i=0]} + \prod_{j=1}^{n} P_{[X_{ij}|Y_i=1]}P_{[Y_i=1]} \end{aligned}$$

If we consider m mutations, we can re-write the likelihood using the number of mutations in the kth category, $N(k)$, which is defined in (4.1). Then, the observed data likelihood is given by

$$L(\theta|N) = \prod_{k=1}^{2^n} C_k (P_{[K(\mathbf{X}_i)=k]})^{N(k)} \tag{4.2}$$

where $C_k = (\frac{N!}{N(x)^{(1)}!\cdots N(x)^{(2^n)}!})$ and $P_{[K(\mathbf{X}_i)=k]} = P_{\mathbf{X}_i}$ using the $\mathbf{X}_i = K^{-1}(k)$ values corresponding to the kth category.

Because maximum likelihood estimation cannot be directly applied with latent variables, we assume we know the latent variable Y_i and employ the Expectation-Maximization algorithm [Dempster et al., 1977]. We note $N(k)$ can be written as

$$N(k) = u(k) + v(k)$$

where

$$\begin{aligned} u(k) &= \sum_{i=1}^{m} I_{[Y_i=0]} I_{[K(\mathbf{X}_i)=k]} \\ v(k) &= \sum_{i=1}^{m} I_{[Y_i=1]} I_{[K(\mathbf{X}_i)=k]} \end{aligned}$$

yielding the complete data likelihood:

$$L_0(\theta|N,v) = \prod_{k=1}^{2^n} C_k [P_{[\mathbf{X}_i|Y_i=0]}P_{[Y_i=0]}]^{u(k)} [P_{[\mathbf{X}_i|Y_i=1]}P_{[Y_i=1]}]^{v(k)} \tag{4.3}$$

using the $\mathbf{X}_i = K^{-1}(k)$ values corresponding to the kth category. We note it is sufficient to define the complete data likelihood in terms of (N,u) or (N,v) because

$u = N - v$ and the observed data likelihood can be written as a summation over all possible $v(k)$.

$$L(\theta|N) = \sum_{v} \binom{N(k)}{v(k)} L_0(\theta|N, v)$$

postMUT Model

We extend postMUT (simple) by introducing a second latent variable $Z = (Z_1, \ldots, Z_m)$ in addition to X_{ij} and Y_i. Z_i corresponds to information about the functionality of the ith mutation that is contained in the protein sequence alignment. Let

$$Z_i = \begin{cases} 1 & \text{if the alignment is informative} \\ 0 & \text{if the alignment is not informative} \end{cases}$$

An example of an informative alignment would be a manually curated alignment containing mostly orthologues (genes derived from speciation events) as opposed to mostly paralogues (genes derived from duplication events). Consider two alignments for the ith mutation from the algorithms: Algorithm A, Algorithm B. Assume both alignments created show mutation i is highly conserved. If Algorithm A produces alignments more 'informative' (e.g. type of homolog used in the alignment) than Algorithm B, then the probability that the mutation will be predicted deleterious by Algorithm A (conditional on Z_i) is higher than the probability the mutation will be predicted deleterious by Algorithm B (conditional on Z_i).

With this in mind, we define a new model which again is a mixture of Bernoulli distributions, but with parameters $\theta = (d_1, \ldots, d_n, e_1, \ldots, e_n, \delta, \gamma, p)$ where

$$d_j = P(X_{ij} = 1|Z_i = 0), \qquad e_j = P(X_{ij} = 1|Z_i = 1)$$

$$\delta = P(Z_i = 0|Y_i = 0), \qquad \gamma = P(Z_i = 1|Y_i = 1)$$

$$p = P(Y_i = 1)$$

Though the addition of the latent variable Z_i does increase the number of parameters, we will show it adds flexibility to estimation. We assume conditional independence between n algorithms given knowledge of Z_i (as opposed to the postMUT (simple) model which assumes conditional independence between algorithms given knowledge of Y_i), then the joint likelihood of $\mathbf{X}_i$ can be written as

$$\begin{aligned} P_{\mathbf{X}_i} = {} & P_{[\mathbf{X}_i|Z_i=0]}P_{[Z_i=0|Y_i=0]}P_{[Y_i=0]} + P_{[\mathbf{X}_i|Z_i=0]}P_{[Z_i=0|Y_i=1]}P_{[Y_i=1]} \\ & + P_{[\mathbf{X}_i|Z_i=1]}P_{[Z_i=1|Y_i=0]}P_{[Y_i=0]} + P_{[\mathbf{X}_i|Z_i=1]}P_{[Z_i=1|Y_i=1]}P_{[Y_i=1]} \end{aligned}$$

We can write the likelihood using the number of mutations in the kth category, $N(k)$ using (4.2) but $P_{[K(\mathbf{X}_i)=k]} = P_{\mathbf{X}_i}$ from the postMUT model using the $\mathbf{X}_i = K^{-1}(k)$ values corresponding with the kth category.

Assuming we know the latent variables Z_i and Y_i, we can write $N(k)$ as

$$N(k) = r(k) + u(k) + v(k) + w(k)$$

where

$$\begin{aligned} r(k) &= \sum_{i=1}^{m} I_{[Z_i=0]}I_{[Y_i=0]}I_{[K(\mathbf{X}_i)=k]} \\ u(k) &= \sum_{i=1}^{m} I_{[Z_i=0]}I_{[Y_i=1]}I_{[K(\mathbf{X}_i)=k]} \\ v(k) &= \sum_{i=1}^{m} I_{[Z_i=1]}I_{[Y_i=0]}I_{[K(\mathbf{X}_i)=k]} \\ w(k) &= \sum_{i=1}^{m} I_{[Z_i=1]}I_{[Y_i=1]}I_{[K(\mathbf{X}_i)=k]} \end{aligned}$$

yielding the complete data likelihood

$$\begin{aligned} L_0(\theta|N,u,v,w) = \prod_{k=1}^{2^n} C_k [P_{[\mathbf{X}_i|Z_i=0]} P_{[Z_i=0|Y_i=0]} P_{[Y_i=0]}]^{r(k)} \\ \times [P_{[\mathbf{X}_i|Z_i=0]} P_{[Z_i=0|Y_i=1]} P_{[Y_i=1]}]^{u(k)} \\ \times [P_{[\mathbf{X}_i|Z_i=1]} P_{[Z_i=1|Y_i=0]} P_{[Y_i=0]}]^{v(k)} \\ \times [P_{[\mathbf{X}_i|Z_i=1]} P_{[Z_i=1|Y_i=1]} P_{[Y_i=1]}]^{w(k)} \end{aligned}$$

using the $\mathbf{X}_i = K^{-1}(k)$ values corresponding to the kth category. We note it is sufficient to define the complete data likelihood in terms of (N,u,v,w) because $r = N-u-v-w$. Similar to the postMUT (simple) model, the observed data likelihood can be written as a sum over all possible $u(k)$, $v(k)$, $w(k)$.

To estimate the sensitivity (b_j) and 1-specificity (a_j) for each jth algorithm, we compute the marginal probabilities

$$a_j = d_j\delta + e_j(1-\delta) \tag{4.4}$$

$$b_j = d_j(1-\gamma) + e_j\gamma \tag{4.5}$$

4.2.3 Parameter Estimation using EM Algorithm

We used the Expectation-Maximization (EM) algorithm [Dempster et al., 1977] to estimate the unknown parameters θ. First, we compute the expectation of the logarithm of the complete data likelihood with respect to $v|N,\theta'$ followed by the maximization with respect to θ.

postMUT (simple) Model

Expectation Step

In Equation 4.3, we provided the complete data likelihood $L_0(\theta|v)$. The logarithm of

the complete data likelihood is given by

$$\begin{aligned}\log L_0(\theta|N,v) &= \log\prod_{k=1}^{2^n} C_k[\prod_{j=1}^{n} P_{[X_{ij}|Y_i=0]}P_{[Y_i=0]}]^{(N(k)-v(k))}[\prod_{j=1}^{n} P_{[X_{ij}|Y_i=1]}P_{[Y_i=1]}]^{v(k)} \\ &= \log(C_K) \\ &\quad + \sum_k[(N(k)-v(k))\{\sum_j x_{ij}\log(a_j)+(1-x_{ij})\log(1-a_j)+\log(1-p)\} \\ &\quad + v(k)\{\sum_j x_{ij}\log(b_j)+(1-x_{ij})\log(1-b_j)+\log(p)\}]\end{aligned}$$

We note

$$E_{v|N,\theta'}[v(k)] = E_{v|N,\theta'}[\sum_{i=1}^{m} I_{[Y_i=1]}I_{[K(\mathbf{x}_i)=k]}] = \sum_{i=1}^{m} P[Y_i=1|N,\theta']I_{[K(\mathbf{x}_i)=k]} = \pi(\mathbf{x}_i)N(k)$$

where

$$\begin{aligned}\pi(\mathbf{x}_i) &= P[Y_i=1|N,\theta'] \\ &= \frac{\prod_{j=1}^{n}(b_j')^{x_{ij}}(1-b_j')^{(1-x_{ij})}p'}{\prod_{j=1}^{n}(a_j')^{x_{ij}}(1-a_j')^{(1-x_{ij})}(1-p')+\prod_{j=1}^{n}(b_j')^{x_{ij}}(1-b_j')^{(1-x_{ij})}p'}\end{aligned}$$

using the $\mathbf{x}_i = K^{-1}(k)$ values associated with the kth category and $\theta' = (a_1', \ldots, a_n', b_1', \ldots, b_n', p')$ which are the parameter estimates at the previous iteration in the EM algorithm. Next, the expectation of the logarithm of the complete data likelihood (Equation 4.3) with respect to $v|N,\theta'$

$$\begin{aligned}Q(\theta|\theta') &= E_{v|N,\theta'}[\log L_0(\theta|N,v)] \\ &= \log(C_K) \\ &\quad + \sum_k(N(k)-E_{v|N,\theta'}[v(k)])\{\sum_j x_{ij}\log(a_j)+(1-x_{ij})\log(1-a_j)+\log(1-p)\} \\ &\quad + E_{v|N,\theta'}[v(k)]\{\sum_j x_{ij}\log(b_j)+(1-x_{ij})\log(1-b_j)+\log(p)\} \\ &= \log(C_K)+\sum_k N(k)[(1-\pi(\mathbf{x}_i))\{\sum_j x_{ij}\log(a_j)+(1-x_{ij})\log(1-a_j)+(1-p)\} \\ &\quad + \pi(\mathbf{x}_i)\{\sum_j x_{ij}\log(b_j)+(1-x_{ij})\log(1-b_j)+p\}]\end{aligned}$$

where $\mathbf{x}_i = K^{-1}(k)$ are values associated with the kth category

Maximization Step

Maximize $Q(\theta|\theta')$ with respect to $\theta = (a_1, \ldots, a_n, b_1, \ldots, b_n, p)$

$$\frac{\partial Q}{\partial p} = -\sum_k N(k)(1-\pi(\mathbf{x}_i))\frac{1}{1-p} + \sum_k N(k)\pi(\mathbf{x}_i)\frac{1}{p}$$

Setting $\frac{\partial Q}{\partial p} = 0$ and solving for p yields

$$\hat{p} = \frac{\sum_k N(k)\pi(\mathbf{x}_i)}{\sum_k N(k)}$$

where $\mathbf{x}_i = K^{-1}(k)$ are values associated with the kth category.

Similarly,

$$\begin{aligned}
\frac{\partial Q}{\partial a_j} &= \sum_k N(k)(1-\pi(\mathbf{x}_i))[\frac{x_{ij}}{a_j} - \frac{(1-x_{ij})}{(1-a_j)}] \\
\frac{\partial Q}{\partial b_j} &= \sum_k N(k)\pi(\mathbf{x}_i)[\frac{x_{ij}}{b_j} - \frac{(1-x_{ij})}{(1-b_j)}]
\end{aligned}$$

Setting $\frac{\partial Q}{\partial a_j} = 0$, $\frac{\partial Q}{\partial b_j} = 0$ and solving for a_j, b_j yields

$$\begin{aligned}
\hat{a}_j &= \frac{\sum_k N(k)(1-\pi(\mathbf{x}_i))x_{ij}}{\sum_k N(k)(1-\pi(\mathbf{x}_i))} \\
\hat{b}_j &= \frac{\sum_k N(k)\pi(\mathbf{x}_i)x_{ij}}{\sum_k N(k)\pi(\mathbf{x}_i)}
\end{aligned}$$

where $\mathbf{x}_i = K^{-1}(k)$ are values associated with the kth category.

Using these parameter estimates $\hat{\theta}$, we compute the posterior probability for the ith mutation being deleterious in the postMUT (simple) model considering all the joint observed functional predictions $\mathbf{X}_i$ from the n algorithms:

$$P(Y_i = 1|\mathbf{X}_i) = \frac{P(\mathbf{X}_i|Y_i = 1)P(Y_i = 1)}{P(\mathbf{X}_i)}$$

postMUT Model

We used the Expectation-Maximization algorithm [Dempster et al., 1977] again for parameter estimation of θ.

Expectation Step

We provided the complete data likelihood $L_0(\theta|u,v,w)$ ($= L_0$ for short). The logarithm of the complete data likelihood is given by

$$
\begin{aligned}
\log L_0 = \log \prod_{k=1}^{2^n} C_K [\prod_{j=1}^{n} P_{[X_{ij}|Z_i=0]} P_{[Z_i=0|Y_i=0]} P_{[Y_i=0]}]^{(N(k)-u(k)-v(k)-w(k))} \\
\times [\prod_{j=1}^{n} P_{[X_{ij}|Z_i=0]} P_{[Z_i=0|Y_i=1]} P_{[Y_i=1]}]^{u(k)} \\
\times [\prod_{j=1}^{n} P_{[X_{ij}|Z_i=1]} P_{[Z_i=1|Y_i=0]} P_{[Y_i=0]}]^{v(k)} \\
\times [\prod_{j=1}^{n} P_{[X_{ij}|Z_i=1]} P_{[Z_i=1|Y_i=1]} P_{[Y_i=1]}]^{w(k)} \\
= \log(C_K) \\
+ \sum_k [(N(k)-u(k)-v(k)-w(k))\{ \\
\times \sum_j x_{ij} \log(d_j) + (1-x_{ij}) \log(1-d_j) + \log(\delta) + \log(1-p)\} \\
+ u(k)\{\sum_j x_{ij} \log(d_j) + (1-x_{ij}) \log(1-d_j) + \log(1-\gamma) + \log(p)\} \\
+ v(k)\{\sum_j x_{ij} \log(e_j) + (1-x_{ij}) \log(1-e_j) + \log(1-\delta) + \log(1-p)\} \\
+ w(k)\{\sum_j x_{ij} \log(e_j) + (1-x_{ij}) \log(1-e_j) + \log(\gamma) + \log(p)\}]
\end{aligned}
$$

We note

$$
\begin{aligned}
E_{u|N,v,w,\theta'}[u(k)] &= \sum_{i=1}^{m} P[Z_i = 0, Y_i = 1|N, \theta^{'}] I_{[K(\mathbf{x}_i)=k]} = \pi_u(\mathbf{x}_i) N(k) \\
E_{v|N,u,w,\theta'}[v(k)] &= \sum_{i=1}^{m} P[Z_i = 1, Y_i = 0|N, \theta^{'}] I_{[K(\mathbf{x}_i)=k]} = \pi_v(\mathbf{x}_i) N(k) \\
E_{w|N,u,v,\theta'}[w(k)] &= \sum_{i=1}^{m} P[Z_i = 1, Y_i = 1|N, \theta^{'}] I_{[K(\mathbf{x}_i)=k]} = \pi_w(\mathbf{x}_i) N(k)
\end{aligned}
$$

where

$$
\begin{aligned}
\pi_u(\mathbf{x}_i) &= P[Z_i = 0, Y_i = 1|N, \theta^{'}] \\
&= \frac{\prod_{j=1}^{n} (d_j^{'})^{x_{ij}} (1 - d_j^{'})^{(1-x_{ij})} (1 - \gamma^{'}) p^{'}}{\prod_{j=1}^{n} (d_j^{'})^{x_{ij}} (1 - d_j^{'})^{(1-x_{ij})} (1 - p^*) + \prod_{j=1}^{n} (e_j^{'})^{x_{ij}} (1 - e_j^{'})^{(1-x_{ij})} p^*} \\
\pi_v(\mathbf{x}_i) &= P[Z_i = 1, Y_i = 0|N, \theta^{'}] \\
&= \frac{\prod_{j=1}^{n} (e_j^{'})^{x_{ij}} (1 - e_j^{'})^{(1-x_{ij})} (1 - \delta^{'}) (1 - p^{'})}{\prod_{j=1}^{n} (d_j^{'})^{x_{ij}} (1 - d_j^{'})^{(1-x_{ij})} (1 - p^*) + \prod_{j=1}^{n} (e_j^{'})^{x_{ij}} (1 - e_j^{'})^{(1-x_{ij})} p^*} \\
\pi_w(\mathbf{x}_i) &= P[Z_i = 1, Y_i = 1|N, \theta^{'}] \\
&= \frac{\prod_{j=1}^{n} (e_j^{'})^{x_{ij}} (1 - e_j^{'})^{(1-x_{ij})} \gamma^{'} p^{'}}{\prod_{j=1}^{n} (d_j^{'})^{x_{ij}} (1 - d_j^{'})^{(1-x_{ij})} (1 - p^*) + \prod_{j=1}^{n} (e_j^{'})^{x_{ij}} (1 - e_j^{'})^{(1-x_{ij})} p^*}
\end{aligned}
$$

with $p^* = (1 - \delta^{'})(1 - p^{'}) + \gamma^{'} p^{'}$ using the $\mathbf{x}_i = K^{-1}(k)$ values associated with the kth category and $\theta^{'} = (d_1^{'}, \ldots, d_n^{'}, e_1^{'}, \ldots, e_n^{'}, \delta^{'}, \gamma^{'}, p^{'})$ which are the parameter estimates at the previous iteration in the EM algorithm. Next, the expectation of the logarithm of the complete data likelihood with respect to $u, v, w|N, \theta^{'}$

$$
\begin{aligned}
Q(\theta|\theta') &= E_{u,v,w|N,\theta'}[\log L_0(\theta|N,u,v,w)] \\
&= \log(C_K) + \sum_k [(N(k) - E_{u|N,\theta'}[u(k)] - E_{v|N,\theta'}[v(k)] - E_{w|N,\theta'}[w(k)]) \\
&\quad \times \{\sum_j x_{ij}\log(d_j) + (1-x_{ij})\log(1-d_j) + \log(\delta) + \log(1-p)\} \\
&\quad + E_{u|N,\theta'}[u(k)]\{\sum_j x_{ij}\log(d_j) + (1-x_{ij})\log(1-d_j) + \log(1-\gamma) + \log(p)\} \\
&\quad + E_{v|N,\theta'}[v(k)]\{\sum_j x_{ij}\log(e_j) + (1-x_{ij})\log(1-e_j) + \log(1-\delta) + \log(1-p)\} \\
&\quad + E_{w|N,\theta'}[w(k)]\{\sum_j x_{ij}\log(e_j) + (1-x_{ij})\log(1-e_j) + \log(\gamma) + \log(p)\}] \\
&= \log(C_K) + \sum_k N(k)[(1 - \pi_u(\mathbf{x}_i) - \pi_v(\mathbf{x}_i) - \pi_w(\mathbf{x}_i)) \\
&\quad \times \{\sum_j x_{ij}\log(d_j) + (1-x_{ij})\log(1-d_j) + \log(\delta) + \log(1-p)\} \\
&\quad + \pi_u(\mathbf{x}_i)\{\sum_j x_{ij}\log(d_j) + (1-x_{ij})\log(1-d_j) + \log(1-\gamma) + \log(p)\} \\
&\quad + \pi_v(\mathbf{x}_i)\{\sum_j x_{ij}\log(e_j) + (1-x_{ij})\log(1-e_j) + \log(1-\delta) + \log(1-p)\} \\
&\quad + \pi_w(\mathbf{x}_i)\{\sum_j x_{ij}\log(e_j) + (1-x_{ij})\log(1-e_j) + \log(\gamma) + \log(p)\}]
\end{aligned}
$$

where $\mathbf{x}_i = K^{-1}(k)$ are values associated with the kth category.

Maximization Step

Maximize $Q(\theta|\theta')$ with respect to $\theta = (d_1, \ldots, d_n, e_1, \ldots, e_n, \delta, \gamma, p)$

$$
\begin{aligned}
\frac{\partial Q}{\partial p} &= -\sum_k N(k)(1 - \pi_u(\mathbf{x}_i) - \pi_v(\mathbf{x}_i) - \pi_w(\mathbf{x}_i))\frac{1}{1-p} + \sum_k N(k)\pi_u(\mathbf{x}_i)\frac{1}{p} \\
&\quad - \sum_k N(k)\pi_v(\mathbf{x}_i)\frac{1}{1-p} + \sum_k N(k)\pi_w(\mathbf{x}_i)\frac{1}{p}
\end{aligned}
$$

Setting $\frac{\partial Q}{\partial p} = 0$ and solving for p yields

$$
\hat{p} = \frac{\sum_k N(k)(\pi_u(\mathbf{x}_i) + \pi_w(\mathbf{x}_i))}{\sum_k N(k)}
$$

where $\mathbf{x}_i = K^{-1}(k)$ are values associated with the kth category.

Similarly,

$$\begin{aligned}
\frac{\partial Q}{\partial \delta} &= \sum_k N(k)(1-\pi_u(\mathbf{x}_i)-\pi_v(\mathbf{x}_i)-\pi_w(\mathbf{x}_i))\frac{1}{\delta} - \sum_k N(k)\pi_v(\mathbf{x}_i)\frac{1}{1-\delta} \\
\frac{\partial Q}{\partial \gamma} &= -\sum_k N(k)\pi_u(\mathbf{x}_i)\frac{1}{1-\gamma} + \sum_k N(k)\pi_w(\mathbf{x}_i)\frac{1}{\gamma}
\end{aligned}$$

Setting $\frac{\partial Q}{\partial \delta} = 0$, $\frac{\partial Q}{\partial \gamma} = 0$ and solving for δ, γ yields

$$\begin{aligned}
\hat{\delta} &= \frac{\sum_k N(k)(1-\pi_u(\mathbf{x}_i)-\pi_v(\mathbf{x}_i)-\pi_w(\mathbf{x}_i))}{\sum_k N(k)(1-\pi_u(\mathbf{x}_i)-\pi_w(\mathbf{x}_i))} \\
\hat{\gamma} &= \frac{\sum_k N(k)\pi_w(\mathbf{x}_i)}{\sum_k N(k)(\pi_u(\mathbf{x}_i)+\pi_w(\mathbf{x}_i))}
\end{aligned}$$

where $\mathbf{x}_i = K^{-1}(k)$ are values associated with the kth category.

Finally,

$$\begin{aligned}
\frac{\partial Q}{\partial d_j} &= \sum_k N(k)(1-\pi_v(\mathbf{x}_i)-\pi_w(\mathbf{x}_i))[\frac{x_{ij}}{d_j} - \frac{(1-x_{ij})}{(1-d_j)}] \\
\frac{\partial Q}{\partial e_j} &= \sum_k N(k)(\pi_v(\mathbf{x}_i)+\pi_w(\mathbf{x}_i))[\frac{x_{ij}}{e_j} - \frac{(1-x_{ij})}{(1-e_j)}]
\end{aligned}$$

Setting $\frac{\partial Q}{\partial d_j} = 0$, $\frac{\partial Q}{\partial e_j} = 0$ and solving for d_j, e_j yields

$$\begin{aligned}
\hat{d_j} &= \frac{\sum_k N(k)(1-\pi_v(\mathbf{x}_i)-\pi_w(\mathbf{x}_i))x_{ij}}{\sum_k N(k)(1-\pi_v(\mathbf{x}_i)-\pi_w(\mathbf{x}_i))} \\
\hat{e}_j &= \frac{\sum_k N(k)(\pi_v(\mathbf{x}_i)+\pi_w(\mathbf{x}_i))x_{ij}}{\sum_k N(k)(\pi_v(\mathbf{x}_i)+\pi_w(\mathbf{x}_i))}
\end{aligned}$$

where $\mathbf{x}_i = K^{-1}(k)$ are values associated with the kth category.

Using these parameter estimates $\hat{\theta}$, the posterior probability can be calculated for the ith mutation being deleterious in the postMUT model considering all the joint observed functional predictions $\mathbf{X}_i$ from the n algorithms:

$$\begin{aligned}
P(Y_i = 1|\mathbf{X}_i) &= \frac{P(\mathbf{X}_i|Y_i=1)P(Y_i=1)}{P(\mathbf{X}_i)} \\
&= \frac{P(\mathbf{X}_i, Z_i=0|Y_i=1)P(Y_i=1) + P(\mathbf{X}_i, Z_i=1|Y_i=1)P(Y_i=1)}{P(\mathbf{X}_i)}
\end{aligned}$$

4.2.4 Confidence Intervals and Wald Confidence Regions

A $100(1-\alpha)\%$ Wald confidence regions [Wald, 1943] of the joint sensitivity and specificity for each jth algorithm (creating an ellipsoid boundary) are also computed using Fisher's information [Shao, 2003] using asymptotic tests based on likelihoods. We note there are two other asymptotic tests discussed in (Likelihood-ratio test, Rao's score test) which are all asymptotically equivalent to using the Wald test to compute the confidence regions.

To calculate a $100(1-\alpha)\%$ confidence interval for θ, Oakes (1999) showed it was sufficient to use the function $Q(\theta|\theta')$ when calculating the observed information matrix where

$$I(\theta) = -\frac{\partial^2 Q}{\partial \theta^2}|_{\theta=\hat{\theta}}$$

yielding the formula

$$[\hat{\theta} - Z_{1-\alpha/2}(\frac{1}{\sqrt{I(\theta)}}), \hat{\theta} + Z_{1-\alpha/2}(\frac{1}{\sqrt{I(\theta)}})]$$

where $\hat{\theta}$ are the parameter estimates of θ.

A $100(1-\alpha)\%$ Wald confidence region using Fisher's information can be calculated using the formulate

$$\{\theta : (\theta - \hat{\theta})^T V^{-1} (\theta - \hat{\theta}) \leq \chi^2_{p,1-\alpha}\}$$

where

$$V^{-1} = -\frac{\partial^2 Q}{\partial \theta \partial \theta^T}|_{\theta=\hat{\theta}}$$

and $\hat{\theta}$ are the parameter estimates of θ.

4.3 Applications of Capture-Recapture Models

In this section, we obtain functional predictions from multiple *in silico* methods with batch-input capabilities for a large number of mutations and apply our postMUT

models to several sets of human mutations (both with and without gold standards). We show our estimates of sensitivity and specificity of the *in silico* methods (without employing a gold standard) match the estimates of sensitivity and specificity when a gold standard is available. Therefore, we are able to account for accuracy of each *in silico* method when combining predictions of missense mutation functionality. In this latter case, specificity is estimated by the frequency of correctly calling neutral variants neutral and sensitivity is estimated by the frequency of correctly calling deleterious variants deleterious. We also provide an interesting application of our postMUT models which estimates the overall proportion of deleterious mutations in a given set using mutations extracted from matched tumor/normal breast cancer genomes.

In this study we used functional predictions from three groups of *in silico* methods:

1. Methods with batch-input capabilities for a large number of mutations (such as in whole exome sequencing) available on the methods website.

   SIFT. [Ng and Henikoff, 2001] We used the SIFT Batch Protein option available on the website. Mutations were classified as neutral and deleterious.

   MutationAssessor. [Reva et al., 2011] We used the web-based version 1.0 with all default settings. Mutations predicted as neutral and low were classified as neutral mutations and mutations predicted as medium and high were classified as deleterious mutations.

   PolyPhen-2 HumDiv (HD). [Adzhubei et al., 2010] We used the PolyPhen-2 classifier model HumDiv as it is the default option, but PolyPhen-2 HumVar is also available on the website. Mutations predicted as neutral were classified as neutral mutations and mutations predicted possibly damaging and probably damaging were classified as deleterious mutations.

2. dbNSFP. [Liu et al., 2011] We used dbNSFP version 2.0 released April 2012 which contains pre-computed predictions from SIFT, PolyPhen-2 HD and HV, LRT and Mutation Taster. Because the default version on the PolyPhen-2 website is HD, we used PolyPhen-2 (HD) for this analysis and omitted the predictions listed from PolyPhen-2 (HV).

3. Four methods described in Hicks et al. (2011). SIFT, PolyPhen-2 (HD), Align-GVGD, MutationAssessor were used on the set of well-characterized mutations. We used the functional predictions from each algorithm using the protein sequence alignment automatically generated by the algorithm or manually curated for the algorithm.

All functional predictions were obtained on a Mac OS X 10.6.8 using Safari 5.1.7 except the predictions obtained for the well-characterized mutations which were previously described in Hicks et al. (2011). Estimation using the EM algorithm was performed on the Shared University Grid at Rice (SUG@R). For more details related to the identifiability of parameters when performing estimation, see McLachlan and Peel (2000).

4.3.1 HumDiv and HumVar

The first two sets of mutations with known functionality we used are referred to as 'HumDiv' and 'HumVar'. HumVar was first described in Capriotti et al. (2006) and HumDiv was described in Adzhubei et al. (2010). HumVar consists of all human-disease causing mutations extracted from UniProtKB and common ($> 1\%$) human mutations without disease annotations. To compare, HumDiv consists of damaging

alleles causing Mendelian diseases and affecting protein function and neutral mutations found comparing human proteins with closely related homologs. Both sets of mutations are freely available on the website provided by the authors of PolyPhen-2 [Adzhubei et al., 2010].

HumVar and HumDiv were used to train and test the nave bayes classifier used by PolyPhen-2 [Adzhubei et al., 2010]. HumVar listed $n = 21151$ neutral and $n = 22196$ deleterious mutations, as opposed to HumDiv which listed $n = 7539$ neutral and $n = 5564$ deleterious mutations. Considering the subset of mutations with predictions obtained directly from the websites by all three *in silico* methods (SIFT, MutationAssessor, PolyPhen-2), HumVar contained $n = 18319$ neutral and $n = 19702$ deleterious mutations and HumDiv contained $n = 6755$ neutral and $n = 4978$ deleterious mutations.

We previously showed in Figure 4.1 the Venn diagrams of the number of mutations predicted deleterious by the three *in silico* methods SIFT, MutationAssessor and PolyPhen-2 when using HumDiv and HumVar data sets, respectively. There is a 62.8% and 60.4% agreement of predicted frequency of deleterious mutations between these methods in HumDiv and HumVar, respectively. Figure 4.2 depicts estimates of the sensitivity and specificity of the *in silico* methods using the HumDiv and HumVar data sets, respectively. Table 4.3 shows our estimates of sensitivity and specificity of the *in silico* methods (estimated from the postMUT model without employing a gold standard).

4.3.2 Well-characterized Mutations

Because HumDiv and HumVar are not considered LSDBs, we also assessed postMUT on a set of mutations extracted from four LSDBs. We previously described this set

Table 4.3 : Sensitivity and specificity estimates of the *in silico* methods (estimated using the postMUT models without a gold standard) and 95% CIs compared to sensitivity and specificity estimates using the 'gold standard. Considered mutations from HumDiv (n = 6755 Neutral, n = 4978 deleterious) and HumVar (n = 18319 Neutral, n = 19702 deleterious). 95% CIs are reported for postMUT model estimates but not postMUT (simple) because standard errors are on the order of 10^{-4}.

Dataset	Mutation Set	Algorithms						% deleterious mutation
		SIFT		MutationAssessor		PolyPhen-2		
		Spec (%)	Sens (%)	Spec (%)	Sens (%)	Spec (%)	Sens (%)	
HumDiv	Gold Standard	91.4	79.2	86.1	85.4	90.2	88.7	42.4
	postMUT (simple)	97.2	88.3	90.9	93.1	93.7	94.6	41.8
	postMUT	87.8	77.7	81.7	82.7	84.0	83.7	40.0
		(86.4-89.2)	(76.5-79.0)	(80.3-83.1)	(81.4-83.9)	(82.5-85.4)	(82.4-85.0)	(38.0-42.0)
HumVar	Gold Standard	80.5	78.6	78.2	84.8	71.1	87.7	51.8
	postMUT (simple)	94.2	87.1	90.0	90.5	83.9	95.4	54.6
	postMUT	84.2	80.0	79.2	88.4	74.2	83.5	53.3
		(82.8-85.7)	(78.9-81.0)	(77.8-80.7)	(87.4-89.4)	(72.8-75.6)	(82.5-84.5)	(51.5-55.0)

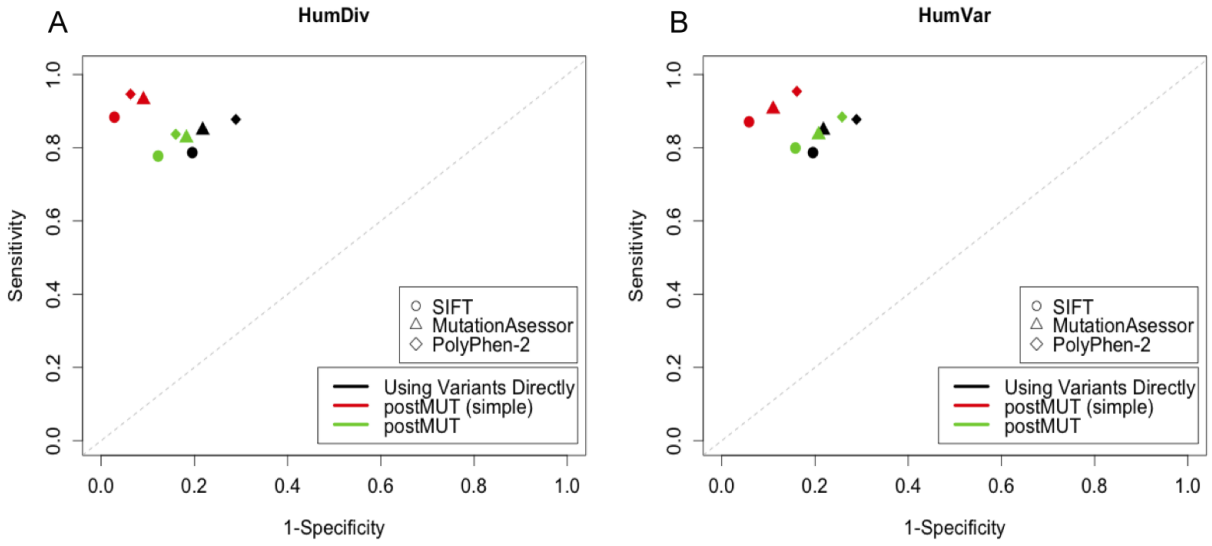

Figure 4.2 : Sensitivity and specificity estimates of the *in silico* methods SIFT, MutationAssessor and PolyPhen-2 (estimated using the postMUT (simple) model (red), the postMUT model (green) without a gold standard) compared to sensitivity and specificity estimated using the gold standard (Using Variants Directly) which means to use the known functional status for each variant (black) in **(A)** HumDiv and **(B)** HumVar.

of mutations and refer to it as well-characterized mutations (Hicks et al. 2011).

The set of mutations referred to as the 'Well-Characterized Data' contains $n = 52$ neutral and $n = 215$ deleterious mutations from four LSDBs considering cancer associated genes: BRCA1, MLH1, MSH2 and TP53. These mutations contain manually curated mutations either using literature or curated locus specific databases for four cancer associated genes. They were previously discussed in Hicks et al. (2011) and Chapter 3 where we used them to compare the accuracy of several algorithms and showed predictions of functionality depend on the algorithm and sequence alignment employed. This data has also been used to compare the accuracy of new

algorithms which predict missense mutation functionality as an independent data set [Acharya and Nagarajaram, 2012, Sim et al., 2012].

In this section, functional predictions for the Well-characterized data were obtained using the pre-computed predictions of four *in silico* methods from dbNSFP (SIFT, PolyPhen-2, LRT and Mutation Taster) and using predictions of four methods obtained directly from websites (SIFT, PolyPhen-2, Align-GVGD and MutationAssessor). Venn diagrams reporting the disjoint categories are not shown for brevity, but there is a 66% agreement of predicted deleterious mutations between the methods using the predictions from dbNSFP and 68.3% agreement between methods obtained directly from websites. We note the functional predictions from PolyPhen-2 and SIFT are not necessarily the same using predictions obtained from dbNSFP compared to predictions obtained from their respective websites. The sensitivity and specificity estimates of the four methods in dbNSFP (Table 4.4) and the four algorithms previously reported in Hicks et al. (2011) (Table 4.5) are reported. In both tables, our postMUT model closely estimates the sensitivity and specificity of the *in silico* methods using sets of mutations extracted from LSDBs.

4.3.3 Matched Normal/Tumor Breast Cancer Sequencing Data

An interesting application of our postMUT models is to investigate the proportion of deleterious mutations using mutations found only in the tumor sample (somatic only) or only in the normal sample (germline only), we obtained missense mutations from two matched normal/tumor breast cancer genomes which are publicly available on the website of Complete Genomics.

We downloaded matched tumor and normal cell line sequence data for two individuals with breast cancer from the Public Genome Data Repository [Drmanac et al., 2010]

Table 4.4 : Sensitivity and specificity estimates of the *in silico* methods (estimated using the postMUT models without a gold standard) and 95% CIs compared to sensitivity and specificity estimates using the 'gold standard. Considered mutations from Well-Characterized (WC) variants using dbNSFP predictions ($n = 50$ Neutral, $n = 264$ deleterious). 95% CIs are reported for postMUT model estimates but not postMUT (simple) because standard errors are on the order of 10^{-4}.

Dataset	Mutation Set	Algorithms								% deleterious mutation
		SIFT		PolyPhen-2		LRT		Mutation Taster		
		Spec (%)	Sens (%)	Spec (%)	Sens (%)	Spec (%)	Sens (%)	Spec (%)	Sens (%)	
WC (dbNSFP)	Gold Standard	50.0	88.8	52.0	88.8	52.0	81.8	48.0	91.6	81.1
	postMUT (simple)	65.2	94.4	73.2	96.1	79.0	90.5	62.4	97.0	78.2
	postMUT	56.2 (55.2-57.2)	91.7 (91.2-92.1)	62.7 (61.6-63.9)	93.0 (92.4-93.5)	68.6 (67.4-69.7)	87.3 (86.8-87.9)	53.5 (52.5-54.4)	94.3 (93.8-94.8)	78.5 (77.5-79.5)

Table 4.5 : Sensitivity and specificity estimates of the *in silico* methods (estimated using the postMUT models without a gold standard) and 95% CIs compared to sensitivity and specificity estimates using the 'gold standard. Considered mutations from Well-Characterized (WC) variants using Hicks et al. 2011 predictions ($n = 52$ Neutral, $n = 215$ deleterious). 95% CIs are reported for postMUT model estimates but not postMUT (simple) because standard errors are on the order of 10^{-4}.

Dataset	Mutation Set	Algorithms								% deleterious mutation
		SIFT		PolyPhen-2		A-GVGD		MutationAssessor		
		Spec (%)	Sens (%)	Spec (%)	Sens (%)	Spec (%)	Sens (%)	Spec (%)	Sens (%)	
WC (Hicks)	Gold Standard	46.2	83.3	53.8	85.1	69.2	84.7	40.4	95.8	80.5
	postMUT (simple)	74.1	92.7	80.3	95.1	86.5	92.6	48.2	98.8	76.5
	postMUT	64.2	89.0	69.1	90.9	74.8	88.2	41.3	96.2	77.5
		((63.2-65.3)	(88.4-89.6)	(68.0-70.4)	(90.2-91.6)	(73.5-76.1)	(87.5-88.9)	(40.5-42.4)	(95.7-96.6)	(76.4-78.7)

publicly available (http://www.completegenomics.com/public-data/) software version 2.0.0.32 and Sample IDs (ATCC Numbers): HCC1187 (CRL-2322), HCC1187 BL (CRL-2323), HCC2218 (CRL-2343), NA12880 (CRL-2363). We filtered for missense mutations with a high variant quality score (VQHigh) resulting in $n = 9236$ and $n = 7487$ mutations in the HCC1187 (normal) and HCC1187 BL (tumor) samples and $n = 9149$ and $n = 8881$ mutations in the HCC2218 (normal) and NA12880 (tumor) samples, respectively. We compared the normal and tumor sets of mutations to obtain mutations in germline only, somatic only or in both the normal and tumor. Using the samples HCC1187 (normal) and HCC1187 BL (tumor), there were $n = 3780$, 2031 and 5456 mutations in germline only, somatic only, in both normal and tumor as compared to using the samples HCC2218 (normal) and NA12880 (tumor) which contained $n = 1467$, 1199 and 7582 mutations in germline only, somatic only, in both normal and tumor. The functional predictions of the mutations from the different sets of mutations from the three *in silico* methods SIFT, MutationAssessor and PolyPhen-2 were obtained directly from their respective websites. The number of mutations with predictions from all three algorithms is reported in Tables 4.6 and 4.7. Using the postMUT (simple) and postMUT models, sensitivity and specificity estimates were calculated for each algorithm in addition to the overall proportion of deleterious mutations. The 1000 Genomes (Feb 2012 release) minor allele frequencies (MAFs) for each set of mutations were obtained from wANNOVAR [Wang et al., 2010, Chang and Wang, 2012].

Using the postMUT (simple) model (Table 4.6) and postMUT model (Table 4.7), we report an enrichment of deleterious mutations in the germline only and somatic only set of mutations compared to the mutations both in the normal and tumor. We calculate the Pearson correlation coefficient between the MAF reported in 1000

Genomes and the continuous score or probability reported by the *in silico* method (SIFT, MutationAssessor, PolyPhen-2) compared to the posterior probability from postMUT and show there is an increase in negative correlation in the germline only and somatic only set of mutations compared to the mutations both in the normal and tumor (Table 4.8). The germline only variants may refer to variants that are on the allele which is lost during tumor development or may be variants where there is not sufficient coverage of the base in the tumor data. However, neither of these two hypotheses is necessarily a unique explanation of why proportion of variants predicted as deleterious would be higher in the germline only variants.

PolyPhen-2 is most correlated with MAF. We calculate the proportion of variants predicted deleterious by each individual algorithm and by postMUT (i.e. at least 2 out of 3 algorithms predicted the mutations deleterious; see Section 4.3.4 for a further discussion on the posterior probabilities) when considering MAF < 0.01 and MAF ≥ 0.01 (Figure 4.3). In general, the proportion of variants predicted deleterious is higher for the rare variants compared to the common variants. PolyPhen-2 has the largest proportion of variants predicted deleterious when considering mutations in both cases (MAF < 0.01 and MAF ≥ 0.01) compared to using the posterior probabilities from postMUT which have a lower proportion of variants predicted deleterious when considering both low-frequency and common variants (MAF ≥ 0.01).

4.3.4 Posterior Probabilities

These posterior probabilities introduced are a function of the estimated parameters (overall proportion of deleterious mutations, sensitivity and specificity of each in silico method) and the functional predictions reported by each *in silico* method. Consider Table 4.2 which gives example labels for the 8 categories from the $n = 3$ methods.

Table 4.6 : Sensitivity and specificity estimates using postMUT (simple) model of the *in silico* methods SIFT, MutationAssessor and PolyPhen-2 (estimated using the postMUT models without a gold standard) considering five different sets of mutations in the matched normal/tumor breast cancer genomes (Sample IDs: HCC1187, HCC2218): normal, tumor, germline only, somatic only and in both normal and tumor. No gold standard is available for these mutations.

Sample ID	Mutation Set	# of Mutations	Algorithms						% deleterious mutation
			SIFT		MutationAssessor		PolyPhen-2		
			Spec (%)	Sens (%)	Spec (%)	Sens (%)	Spec (%)	Sens (%)	
HCC1187	Normal	7380	96.3	73.5	97.0	63.4	91.4	84.3	16.6
	Tumor	5967	96.2	75.4	97.3	63.1	91.8	82.4	16.4
	Germline only	3041	96.0	73.0	96.1	62.6	88.5	88.7	19.6
	Somatic only	1628	95.3	77.3	96.8	61.3	97.6	86.0	21.7
	Both	4339	96.4	74.4	97.5	64.4	93.2	80.3	14.4
HCC2218	Normal	7388	96.4	70.9	97.6	63.3	91.8	83.4	17.4
	Tumor	7210	96.4	71.6	97.5	64.6	91.6	83.9	17.7
	Germline only	1107	94.7	71.8	97.4	63.2	88.5	87.3	21.8
	Somatic only	929	94.8	75.6	96.7	69.6	86.0	89.3	24.7
	Both	6281	96.6	70.6	97.6	63.4	92.4	82.5	16.7

Table 4.7 : Sensitivity and specificity estimates using postMUT model of the *in silico* methods SIFT, MutationAssessor and PolyPhen-2 (estimated using the postMUT models without a gold standard) considering five different sets of mutations in the matched normal/tumor breast cancer genomes (Sample IDs: HCC1187, HCC2218): normal, tumor, germline only, somatic only and in both normal and tumor. No gold standard is available for these mutations.

Sample ID	Mutation Set	# of Mutations	Algorithms						% deleterious mutation
			SIFT		MutationAssessor		PolyPhen-2		
			Spec (%)	Sens (%)	Spec (%)	Sens (%)	Spec (%)	Sens (%)	
HCC1187	Normal	7380	92.6	65.3	93.7	56.4	87.4	75.4	13.5
	Tumor	5967	92.7	67.1	94.4	56.1	88.2	73.7	13.7
	Germline only	3041	91.7	65.3	92.4	56.1	83.6	80.2	15.9
	Somatic only	1628	90.0	69.6	92.5	55.1	82.1	78.2	17.2
	Both	4339	92.9	65.4	94.5	56.5	89.6	71.0	11.4
HCC2218	Normal	7388	92.6	63.5	94.2	56.7	87.6	75.2	14.1
	Tumor	7210	92.7	63.9	94.1	57.6	87.5	75.3	14.6
	Germline only	1107	90.4	64.6	93.6	56.6	83.6	79.0	18.4
	Somatic only	929	89.7	64.7	91.9	59.4	80.6	77.7	22.6
	Both	6281	93.3	63.5	94.6	57.0	88.7	74.6	13.8

Table 4.8 : The correlation between 1000 Genomes minor allele frequency (MAF) and the continuous score or probability reported by the *in silico* methods (SIFT, MutationAssesor, PolyPhen-2) and postMUT posterior probability.

Sample ID	Mutation Set	Correlation			
		SIFT	MutationAssessor	PolyPhen-2	postMUT
HCC1187	Normal	0.070	-0.082	-0.112	-0.100
	Tumor	0.053	-0.054	-0.103	-0.096
	Germline only	**0.218**	**-0.257**	**-0.238**	**-0.201**
	Somatic only	**0.257**	**-0.269**	**-0.291**	**-0.251**
	Both	-0.032	0.033	-0.009	-0.019
HCC2218	Normal	0.062	-0.082	-0.115	-0.111
	Tumor	0.068	-0.094	-0.128	-0.119
	Germline only	**0.188**	**-0.206**	**-0.215**	**-0.223**
	Somatic only	**0.159**	**-0.214**	**-0.231**	**-0.207**
	Both	0.037	-0.056	-0.091	-0.085

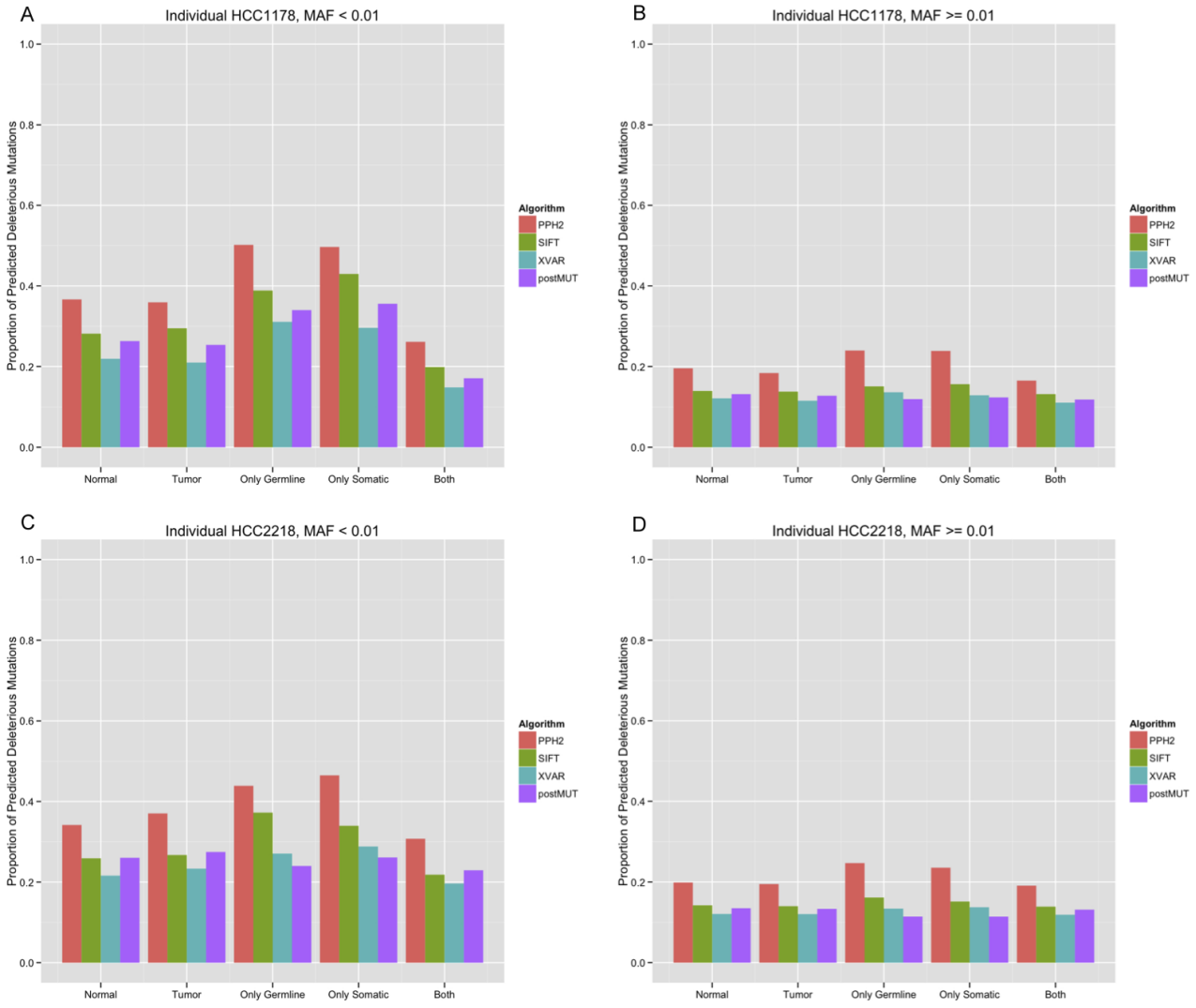

Figure 4.3 : The proportion of mutations predicted deleterious by the three *in silico* methods (PolyPhen-2 (PPH2), SIFT, MutationAssessor (Xvar)) and by postMUT (high posterior probability) considering set of mutations that are rare (MAF < 0.01) or low-frequency and common (MAF ≥ 0.01) for individuals HCC1187 **(A)**, **(B)** and HCC2218 **(C)**, **(D)**.

Figure 4.4 gives example posterior probabilities for these $n = 3$ methods when the overall proportion of deleterious mutations is small (20%) and large (80%), respectively. For each figure, we consider two models differing if all the methods perform similarly in accuracy or not. In Model A, 2 out of the 3 methods perform similarly well, but the third performs worse resulting in posterior probabilities that weigh the first two methods more heavily than the third algorithm. In Model B and all three methods perform similarly well resulting in posterior probabilities with a wide gap between Group 2 (only one method predicted the mutation deleterious) and Group 3 (at least two methods predicted the mutation deleterious). We also see the posterior probabilities are heavily influenced by the overall proportion of deleterious mutations as seen by the overall shift between Figure 4.4A and Figure 4.4B.

4.3.5 Identifying Functional Mutations in Whole-Exome Sequencing Data

We applied the postMUT models to unpublished real human exome sequencing data obtained from a collaborator Sharon Plon, M.D., Ph.D., at Baylor College of Medicine in Houston, Texas. This data was initially sequenced to identify cancer susceptibility genes for acute lymphocytic leukemia (ALL) and lymphoma including probands with childhood onset of leukemia and siblings (and more extended relatives) with ALL or lymphoma. ALL is the most common pediatric cancer, but little is known about the inherited predisposition to ALL. Figure 4.5 describes an extended pedigree with four cases of lymphocytic leukemia and lymphoma for which samples and cell lines are available on multiple family members. Whole exome sequencing analysis was performed at the Human Genome Sequencing Center (HGSC) at Baylor College of Medicine under the direction of David Wheeler, Ph.D. and Richard Gibbs, Ph.D.

Using the postMUT model, the overall proportion of deleterious mutations esti-

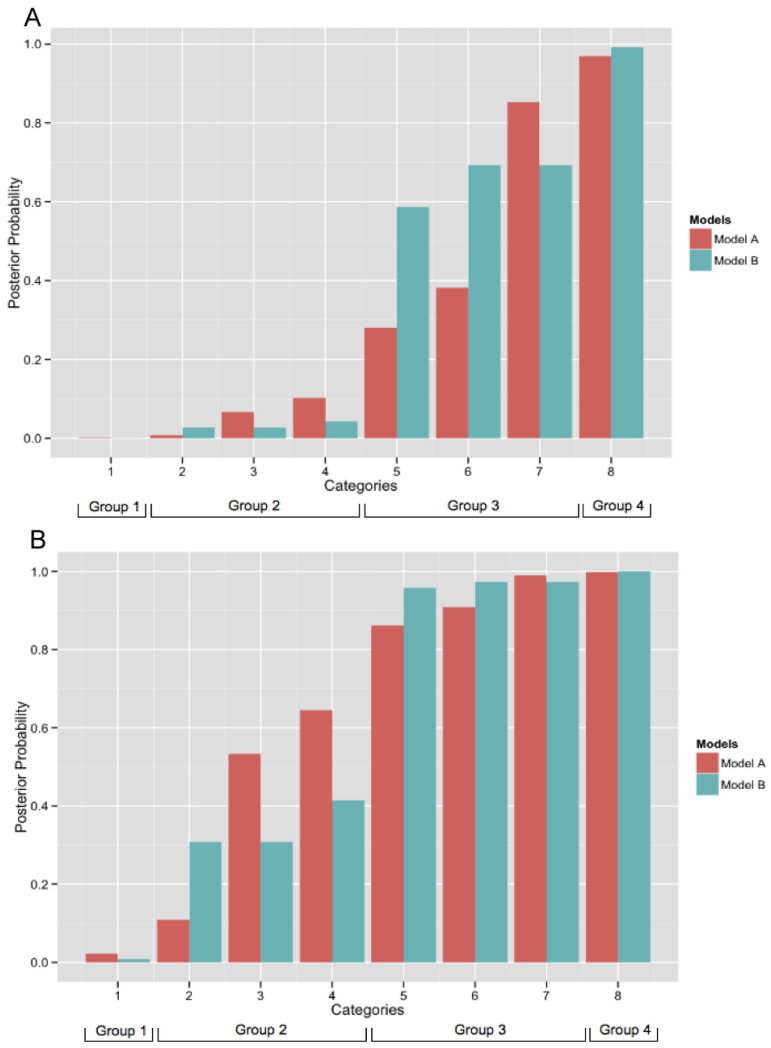

Figure 4.4 : Example posterior probabilities as a function of sensitivity and specificity of $n = 3$ *in silico* methods with 8 categories where the overall proportion of deleterious mutations is $p = 0.20$ in **(A)** and $p = 0.80$ in **(B)**. In Model A the first two methods perform similarly well ($a_1 = 0.10$, $a_2 = 0.15$ and $b_1 = 0.90$, $b_2 = 0.90$), but the third method performs worse ($a_3 = 0.30$, $b_3 = 0.70$) as opposed to Model B in which all three methods perform similarly well ($a_1 = 0.10$, $a_2 = 0.15$, $a_3 = 0.10$ and $b_1 = 0.90$, $b_2 = 0.90$, $b_3 = 0.85$). Considering Table 4.2, Group 1 represents category 1 (no algorithms predicted the mutation deleterious), Group 2 represents categories 2-4 (only one algorithm predicted the mutation deleterious), Group 3 represents categories 5-7 (at least two algorithms predicted the mutation deleterious) and Group 4 represents category 8 (all three algorithms predicted the mutation deleterious)

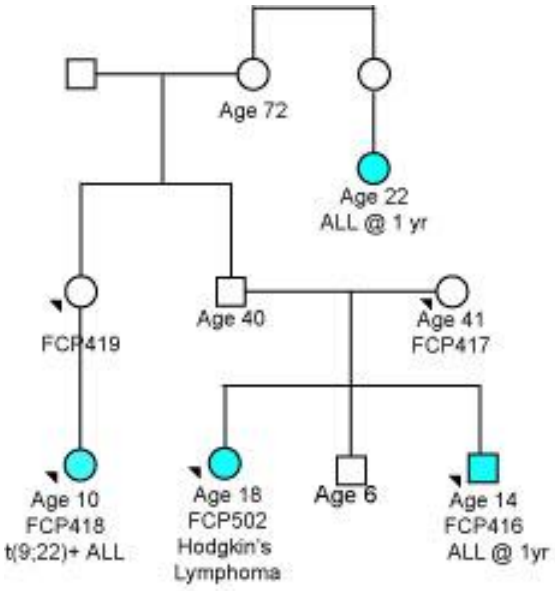

Figure 4.5 : Extended Pedigree with four cases of lymphocytic leukemia and lymphoma

mated for the affected individuals (FCP416, FCP418, FCP502) and unaffected individual (FCP417) was 19.6%, 17.7%, 21.9% and 19.9%, respectively. Table 4.9 shows the posterior probabilities for each variant being deleterious within the individual showed if at least 2 out of 3 algorithms (SIFT, PolyPhen-2 and MutationAssessor) predict the variant to be deleterious, then there is a high posterior probability of being deleterious ($> 80\%$). Because the sensitivity and specificity of the *in silico* methods estimated in this application perform similarly, we use this as a cutoff to determine which mutations are predicted deleterious by the postMUT models.

We applied a set of three filters to search for candidate mutations:

1. Mutations observed in three affected individuals, but not in the unaffected individual

Table 4.9 : Example posterior probabilities of being deleterious given the observed functional prediction from the three algorithms SIFT, MutationAssessor and PolyPhen-2 for an affected (FCP416) and unaffected (FCP417) individual. N = neutral prediction, D = deleterious prediction.

SIFT	MutationAssessor	PolyPhen-2	Posterior (affected)	Posterior (unaffected)
D	D	D	0.932	0.887
N	D	D	0.858	0.836
D	N	D	0.865	0.832
D	D	N	0.845	0.826
D	N	N	0.194	0.211
N	D	N	0.180	0.220
N	N	D	0.218	0.234
N	N	N	0.035	0.048

2. Low minor allele frequency: $< 1\%$

3. High posterior probability from the postMUT model: cutoff $> 80\%$

Using these three filters, we identified a set of 80 candidate missense mutations to be further analyzed using biological functional assays.

4.4 Discussion

Interpreting the functional impact of missense mutations on protein function is an important step in identifying disease-causing mutations. Even though the majority of *in silico* methods predicting the functionality of missense mutations utilize similar information in the form of evolutionary conservation (specifically phylogenetic information) to assess the effect of the mutation on protein function [Jordan et al., 2010], they often lead to conflicting results leaving the user without guidance in assessing the pathogenic impact of missense mutations on protein function. Two recent reviews

[Sifrim et al., 2012, Lyon and Wang, 2012] comparing the features and limitations of these methods concluded first-principles methods (and especially genomic annotation tools) often have disagreeing and constantly evolving annotations leading to different sets of final candidate variants when employing different *in silico* methods.

There are many other technical problems related to using these first-principles and trained classifiers *in silico* methods. For example, most sets of mutations with known functionality (gold standards) used to calibrate or train the classifiers are only weakly associated with disease. Many web-based algorithms do not allow the user to submit a large number of mutations, but some may allow the user to download and maintain an in-house version of the algorithm leading to questions of reproducibility. Wong et al. (2011) discussed the problem of different methods requiring different input and output formats. One way to avoid this problem is to employ databases containing pre-computed predictions of functionality such as dbNSFP [Liu et al., 2011] or SNVBox [Wong et al., 2011], but the pre-computed predictions between different versions of the databases may vary greatly and often the missense mutations are missing from the database entirely depending on the gene list used to create the database. Further research is needed to address these technical difficulties.

These posterior probabilities produced by our postMUT models combine discordant functional predictions from the individual *in silico* methods in the absence of a gold standard by taking advantage of the fact that these methods disagree. Though the parameter estimates from the two postMUT models converge to estimates similar to estimates using a gold standard, issues may exist with identifiability of the parameters when performing estimation. These problems are recognized in a more general context of mixture models [McLachlan and Peel, 2000]. Parameters of our models are estimated using the EM algorithm which iteratively finds the maximum

likelihood when there is latent or missing data involved. Potential problems with the identifiability of parameters when performing estimation are addressed by imposing intuitive identifiability constraints in the parameter space as suggested by McLachlan and Peel (2000). Specifically, we require the false positive rate $= a_j < 0.50$ and true positive rate $= b_j > 0.50$ to prevent the algorithm from yielding predictions worse than fair coin flipping.

Recently, several groups have taken a more comprehensive approach developing annotation and prioritization tools. Functional annotation tools are the third group of *in silico* methods with the goal *to infer the functionality of missense mutations* by annotating batch sets of mutations from whole-exome or whole-genome data. These are often open-access tools or webservers which report predictions of functionality from different algorithms and annotations from large databases without any formal interpretation of functionality. Conversely, disease gene prioritization tools try *to infer candidate disease genes*. These tools include some functional annotations, but they are mostly focused on automating the filtering process when searching for disease causal genes. Lyon and Wang (2012) describe two approaches on inferring candidate genes from whole-exome or whole-genome sequencing data:

1. A probabilistic scoring approach (conceptually more sophisticated and less likely to miss causal genes) combining multiple sources of data into a statistical model which ranks the genes by their probability of being disease-causing

2. A stepwise reduction approach (also referred to as intersection filtering [Robinson et al., 2011]) which is more easily interpreted removing variants based on filters such as allele frequency, segregation with disease and functional predictions to end up at a list of candidate genes

Some examples of probabilistic scoring disease gene prioritization tools have been developed such as VAAST [Yandell et al., 2011] or KGGSeq [Li et al., 2012], but these tools do not have the same goal as the *in silico* methods inferring the functionality of missense mutations and therefore are not directly comparable. Rather, we argue the posterior probability of pathogenicity used to infer the functionality of missense mutations introduced in this chapter may be incorporated into tools such as in disease gene prioritization tools.

In this chapter, we developed a method to combine these discordant functional predictions and to provide a unifying probability of pathogenicity for each missense mutation which may be used to prioritize the mutations identified for further evaluation in biological laboratory-based assays. As the interpretation of missense mutations still remains an elusive and difficult task, we provide a novel, scalable tool to infer the functionality of missense mutations which is an important step in assessing causality and identifying disease susceptibility mutations.

Chapter 5

Simulation Studies to Evaluate the Performance of postMUT Models

In this chapter, we investigate the performance of the postMUT (simple) and postMUT models by performing simulation studies. We simulate predictions of functionality for m mutations from n *in silico* algorithms using either the postMUT (simple) model (Section 5.1) and the postMUT model (Section 5.2) with known parameters θ. After the obtaining the simulated functional predictions, $\mathbf{X}$, we perform parameter estimation using both postMUT models to obtain $\hat{\theta}$. We assess bias and root mean squared error (RMSE) of the parameters as a function of the number of mutations m. Bias is defined as

$$Bias(\hat{\theta}) = E[\hat{\theta} - \theta] \tag{5.1}$$

and the estimator is said to be unbiased if the bias is equal to zero. MSE is defined as

$$MSE(\hat{\theta}) = E[(\hat{\theta} - \theta)^2] = Var(\hat{\theta}) + [Bias(\hat{\theta})]^2 \tag{5.2}$$

which assess the quality of the estimator in terms of its variance and degree of bias [Shao, 2003].

In both Section 5.1 and 5.2, we asked the following three questions:

1. How do bias and RMSE vary as a function of the overall proportion of deleterious mutations p? (Simulation Study 1)

2. How do bias and RMSE vary as the sensitivity and specificity of *in silico* algorithms changes? (Simulation Study 2)

3. How do bias and RMSE vary as the number of *in silico* algorithms increases? (Simulation Study 3)

5.1 Performance on Simulated Data using the postMUT (simple) Model

Using functional predictions simulated from the postMUT (simple) model, we consider three different simulation studies which are described in Table 5.1. We explore different ranges of the parameters. Parameter estimation is performed using the postMUT (simple) and postMPUT models.

5.1.1 Simulation Study 1: Varying p

As a function of the overall proportion of deleterious mutations p, we consider $p = 0.05, 0.20, 0.80$ to explore the range of $p \in [0, 1]$. When p is small ($p = 0.05$ or 0.20), the RMSE(a_j) $<$ RMSE(b_j) $\forall\ j$ because the majority of the variants in this set are neutral. When using a set of mostly deleterious variants ($p = 0.80$), the RMSE(a_j) $>$ RMSE(b_j) $\forall\ j$ because the majority of the variants are deleterious (Figures 5.1, 5.2). Figure 5.3 shows example asymptotic Wald confidence regions for increasing false positive rates ($a_1 = 0.10 < a_2 = 0.20 < a_3 = 0.30$) and decreasing true positive rates are decreasing ($b_1 = 0.10 > b_2 = 0.20 > b_3 = 0.30$) when considering a small proportion of variants, $p = 0.20$ and large proportion of variants, $p = 0.80$ with $n = 3$ algorithms and $m = 2000$ mutations. The confidence regions using $p = 0.20$ are ellipsoids growing larger along the y-axis as sensitivity (true positive rate) decreases

Table 5.1 : Parameters used for simulation studies described Section 5.1

Simulation Simulation Studys		a_1	a_2	a_3	a_4	b_1	b_2	b_3	b_4	p
	Model A	0.15	0.15	0.15	-	0.90	0.90	0.90	-	0.20
	Model A									
Simulation Study 1: Varying p	Model 1.1	0.15	0.15	0.15	-	0.90	0.90	0.90	-	**0.05**
	Model 1.2	0.15	0.15	0.15	-	0.90	0.90	0.90	-	**0.80**
	Model A									
	Model 2.1	0.15	0.15	**0.25**	-	0.90	0.90	**0.75**	-	0.20
Simulation Study 2: Varying a_j, b_j	Model 2.2	0.15	**0.25**	**0.25**	-	0.90	**0.75**	**0.75**	-	0.20
	Model 2.3	**0.25**	**0.25**	**0.25**	-	**0.75**	**0.75**	**0.75**	-	0.20
	Model A									
	Model 3.1	0.15	0.15	0.15	**0.15**	0.90	0.90	0.90	**0.90**	**0.20**
Simulation Study 3: Varying n	Model 1.2									
	Model 3.2	0.15	0.15	0.15	**0.15**	0.90	0.90	0.90	**0.90**	**0.80**

because the majority of the mutations are neutral and therefore less information is available to estimate the sensitivity. Similarly, using $p = 0.80$, the ellipsoid confidence regions grow larger along the x-axis as 1 - specificity (false positive rate) grows larger because most of the mutations are deleterious. We note the confidence regions are also a function of the number of mutations: as the number of mutations increase, the confidence regions decrease. Additionally, when p is small $\hat{p}$ is negatively biased, but positively biased when $\hat{p}$ is large (Figure 5.1) when estimating with the postMUT model.

5.1.2 Simulation Study 2: Varying a_j, b_j

Next, we compare models which vary a_j and b_j. As we increase the false positive rate and decrease the true positive rate (Model A, Model 2.1, Model 2.2, Model 2.3 in Table 5.1), we see the bias and RMSE increases (Figures 5.4, 5.5). Surprisingly, when comparing Model A to Model 2.1 we see the $\text{MSE}(a_1) = \text{RMSE}(a_2) > \text{RMSE}(a_3)$ and $\text{RMSE}(b_1) = \text{RMSE}(b_2) > \text{RMSE}(b_3)$ even though the sensitivity and specificity of algorithms 1 and 2 are better than algorithm 3. This seems to stem from the nature of the disjoint categories. When the algorithms perform equally well in sensitivity and specificity (Model A), the algorithms disagree equally. When the algorithms do not perform equally well (Model 2.1), the algorithms do not disagree equally forcing some categories to be weighed more heavily (Figure 5.6) influencing the accuracy of the postMUT models.

5.1.3 Simulation Study 3: Varying n

As the number of algorithms increases, the number of parameters to estimates increases in both the postMUT (simple) model ($2n + 1$) and the postMUT model ($2n$

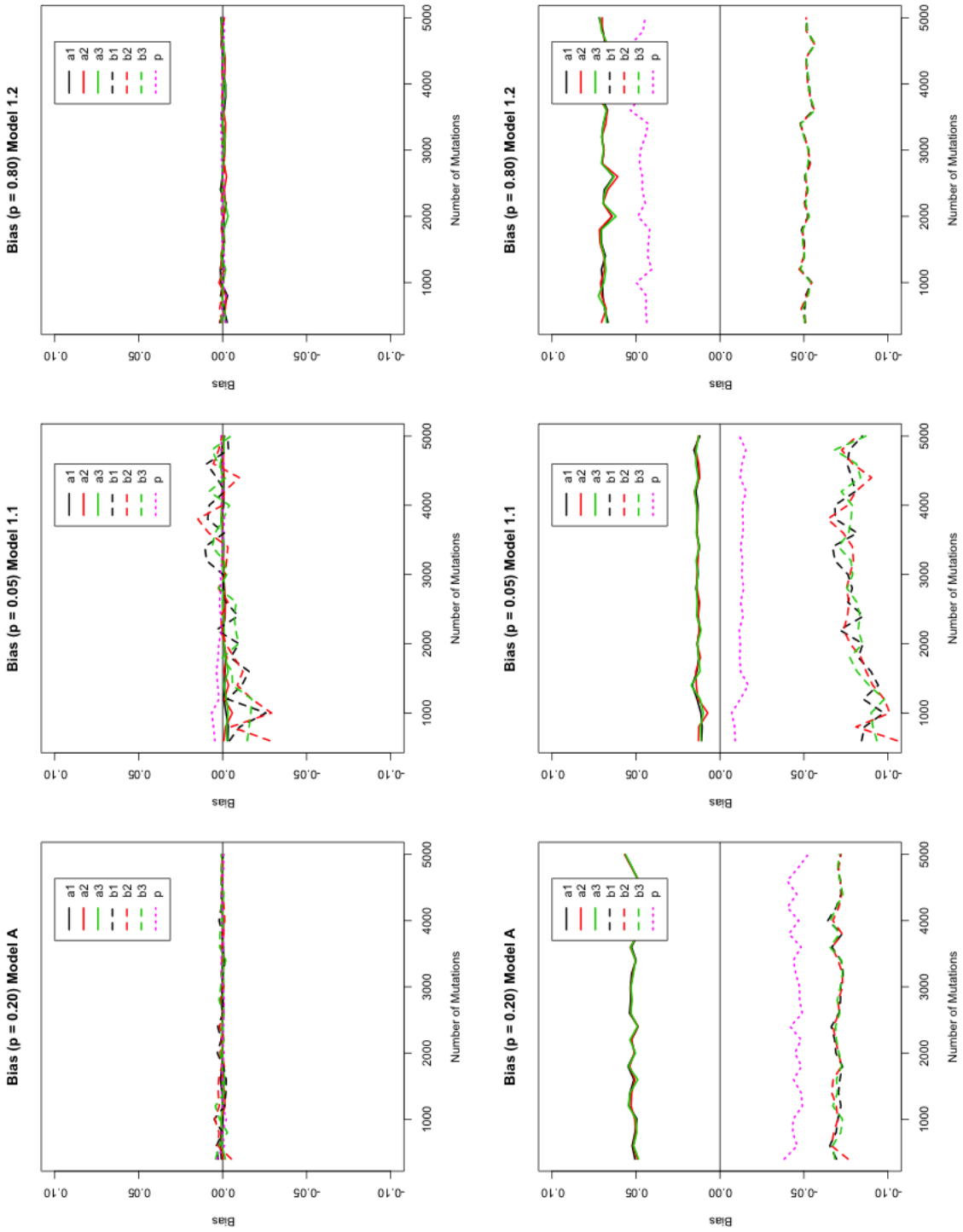

Figure 5.1 : Simulation Study 1: Varying p (Table 5.1). Assessing Bias using $p = 0.20$, $p = 0.05$ and $p = 0.80$. Row 1: Estimated with postMUT (simple), Row 2: Estimated with postMUT.

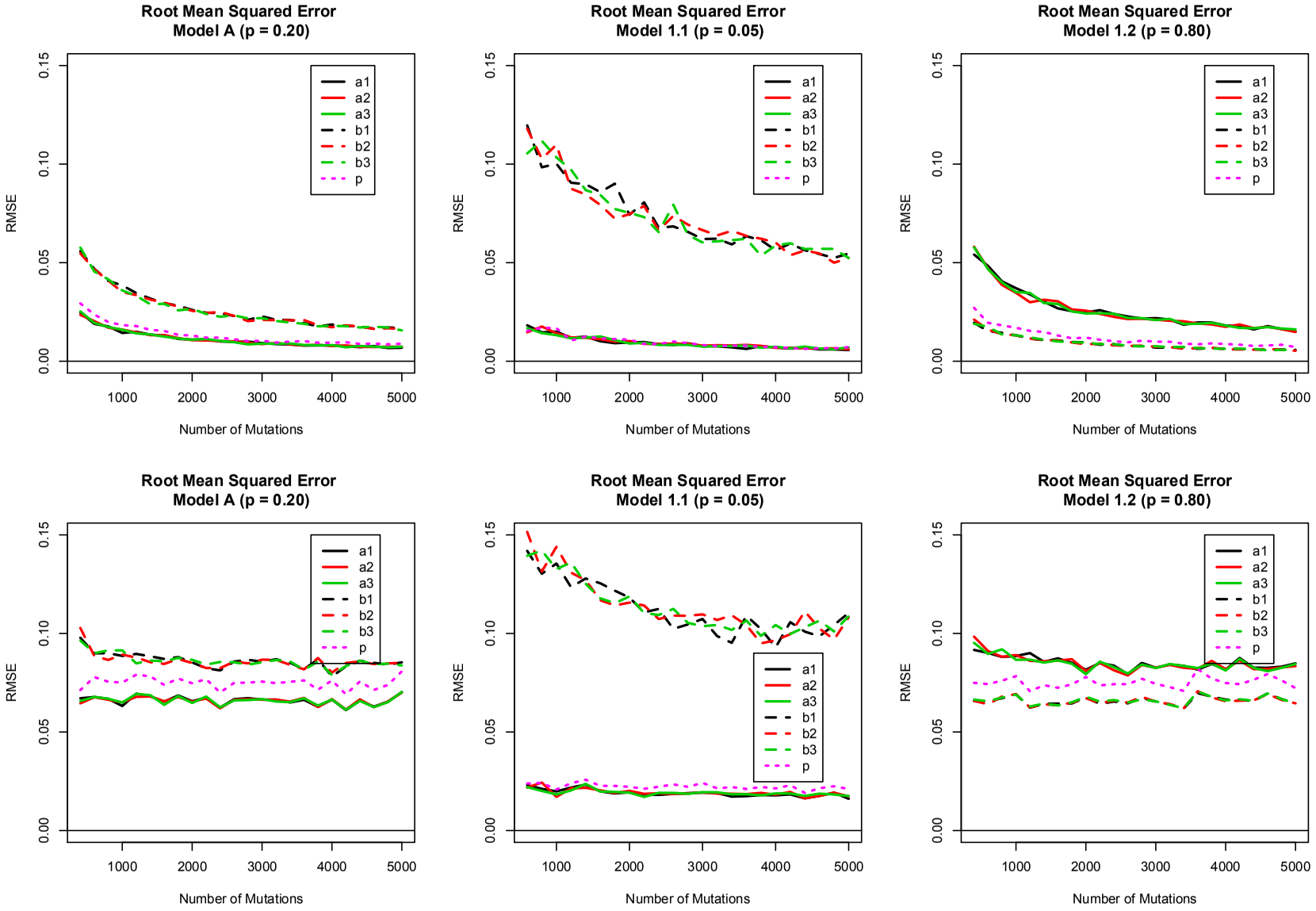

Figure 5.2 : Simulation Study 1: Varying p (Table 5.1). Assessing RMSE using $p = 0.20$, $p = 0.05$ and $p = 0.80$. Row 1: Estimated with postMUT (simple), Row 2: Estimated with postMUT.

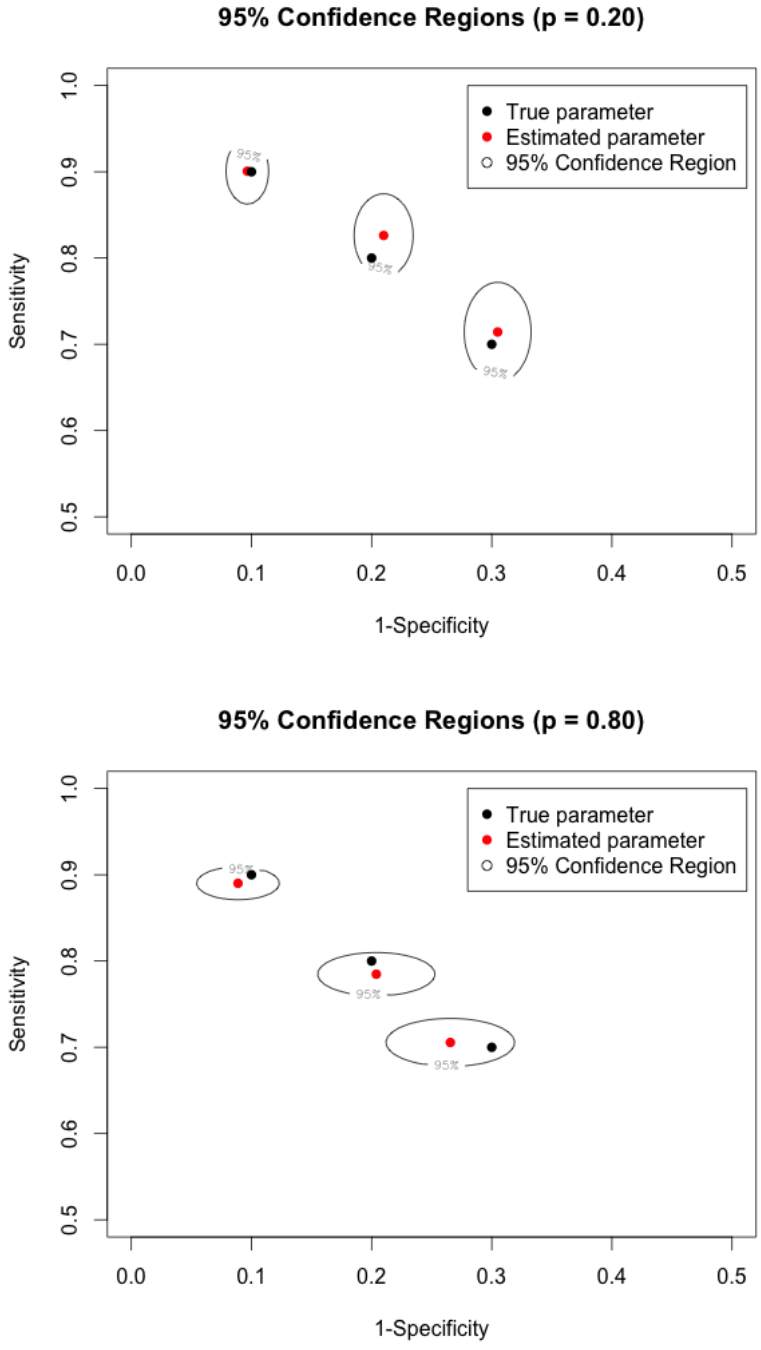

Figure 5.3 : Examples of 95% Wald confidence regions of sensitivity and specificity parameter estimates using postMUT (simple) model considering $n = 3$ simulated *in silico* algorithms, $m = 2000$ mutations and $p = 0.20$ (left), $p = 0.80$ (right). For each of the $n = 3$ algorithms, mutations were simulated with increasing false positive rates ($a_1 = 0.10 < a_2 = 0.20 < a_3 = 0.30$) and decreasing true positive rates ($b_1 = 0.90 > b_2 = 0.80 > b_3 = 0.70$).

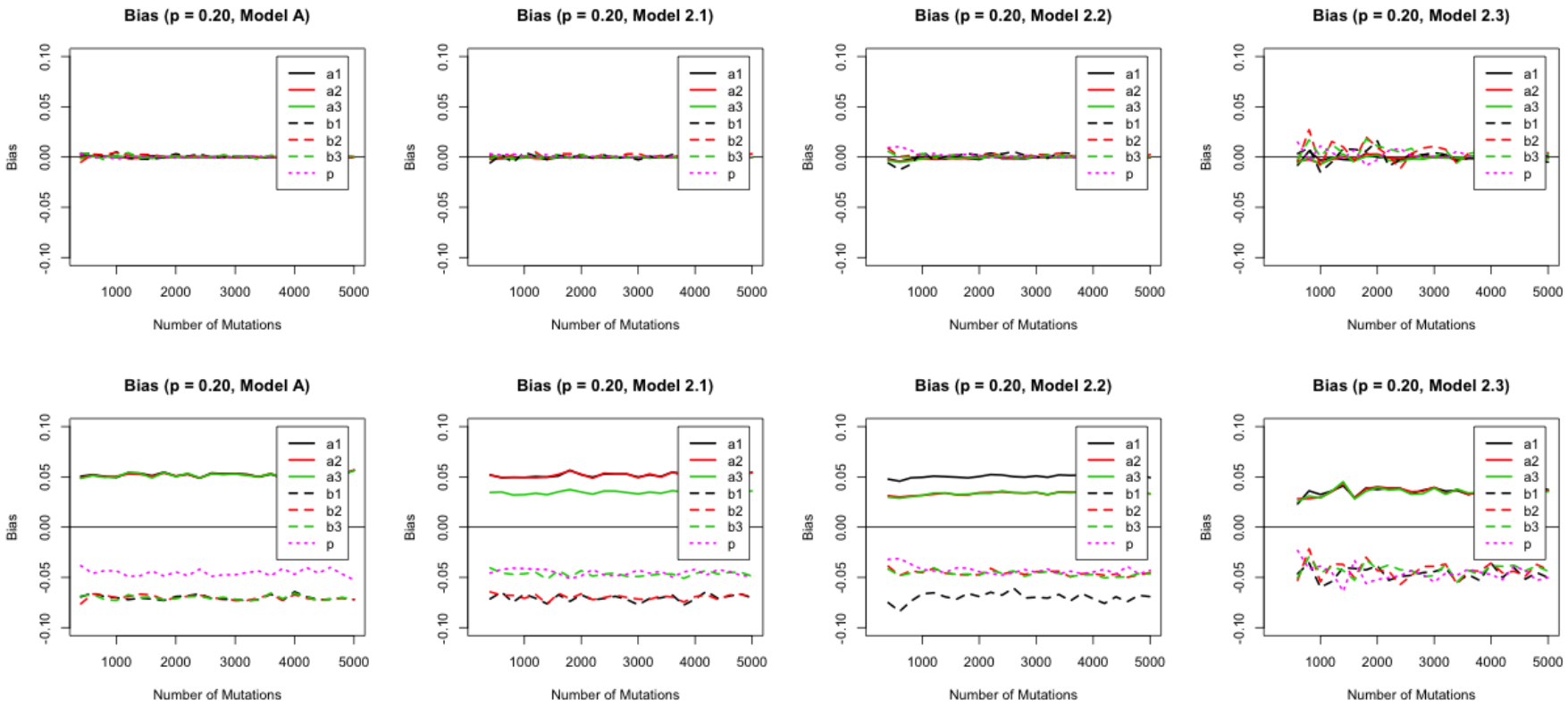

Figure 5.4 : Simulation Study 2: Varying a, b (Table 5.1). Assessing Bias.
Row 1: Estimated with postMUT (simple), Row 2: Estimated with postMUT.

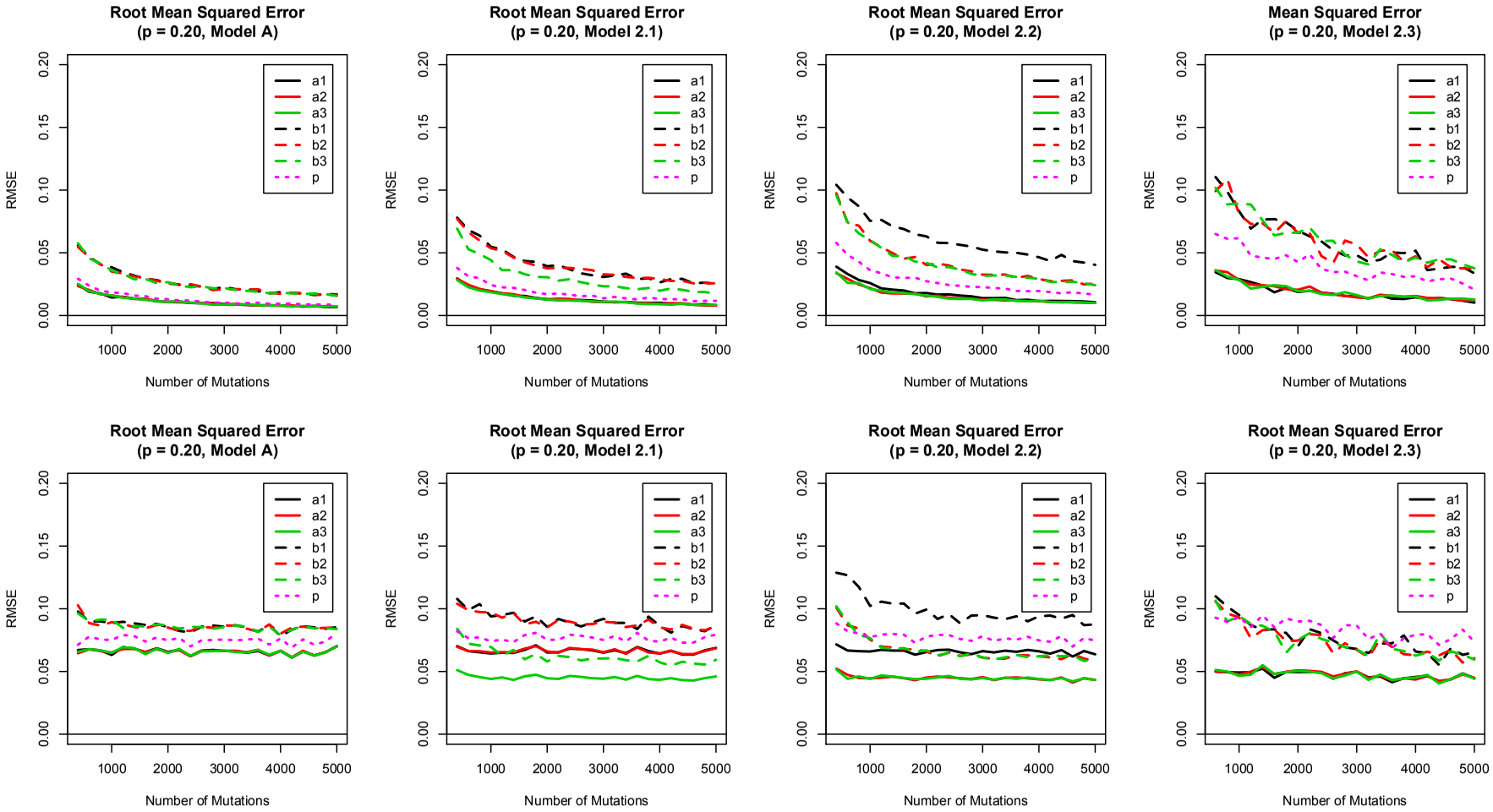

Figure 5.5 : Simulation Study 2: Varying a, b (Table 5.1). Assessing RMSE
Row 1: Estimated with postMUT (simple), Row 2: Estimated with postMUT.

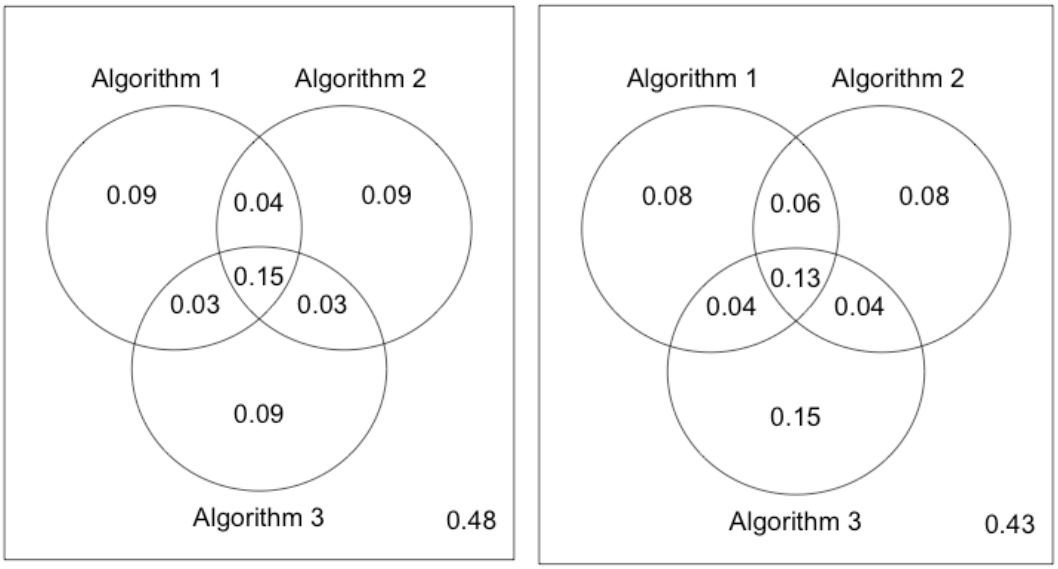

Figure 5.6 : Venn diagrams comparing the proportion of mutations ($m = 2000$) simulated from the postMUT (simple) model when $n = 3$ *in silico* algorithms perform equally well in sensitivity and specificity (Model A in Table 2, left Venn diagram) and when the *in silico* algorithms do not perform equally well (Model 2.1 in Table 2, right Venn diagram).

+ 3). Even with more parameters to estimate, the overall RMSE decreases when $n =$ 4 than compared to $n = 3$ in both the cases $p = 0.20$ and $p = 0.80$ (Figures 5.7, 5.8). This indicates it is advantageous to use functional predictions from more algorithms.

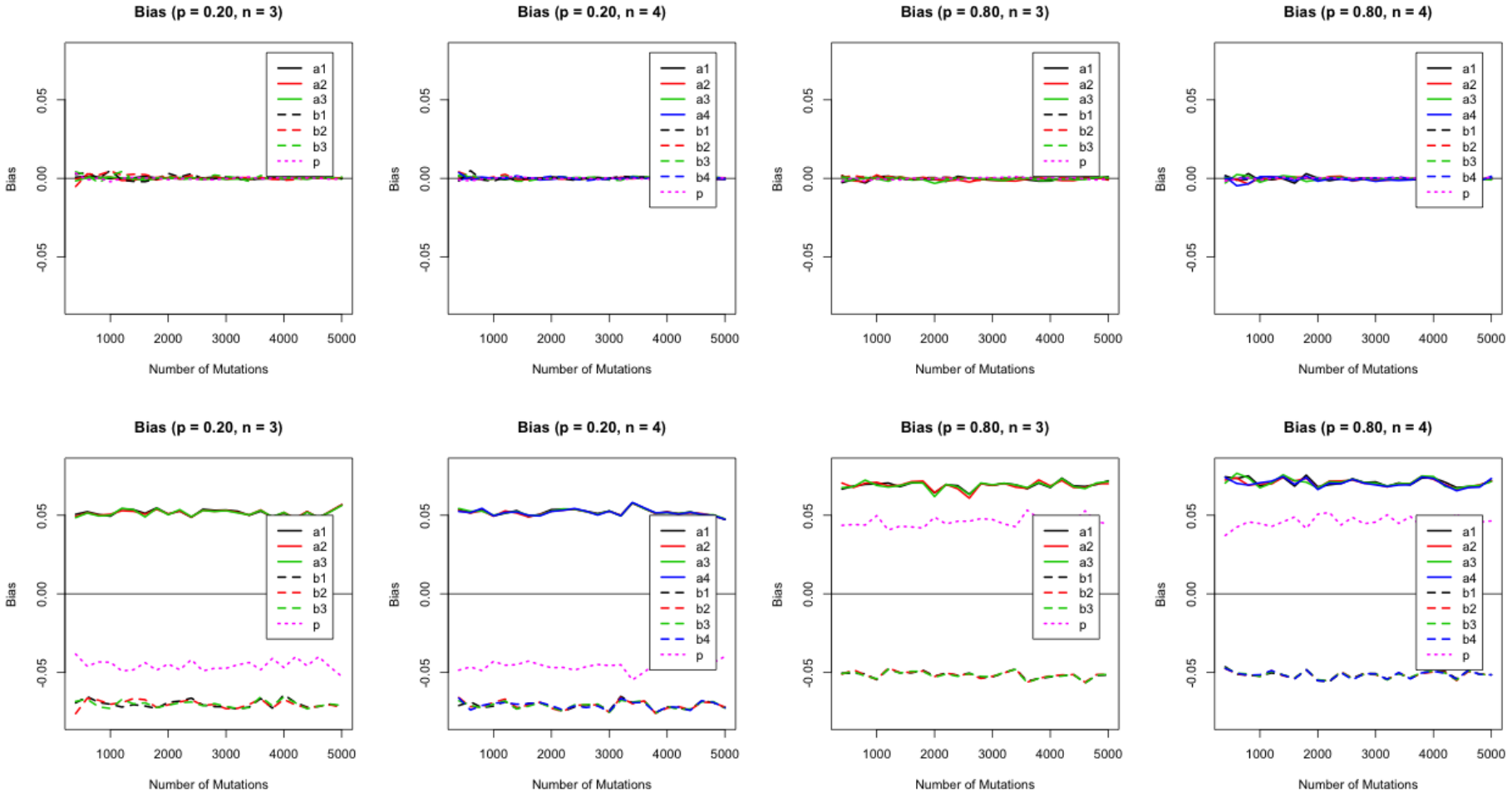

Figure 5.7 : Simulation Study 3: Varying n (Table 5.1). Assessing Bias.
Row 1: Estimated with postMUT (simple), Row 2: Estimated with postMUT.

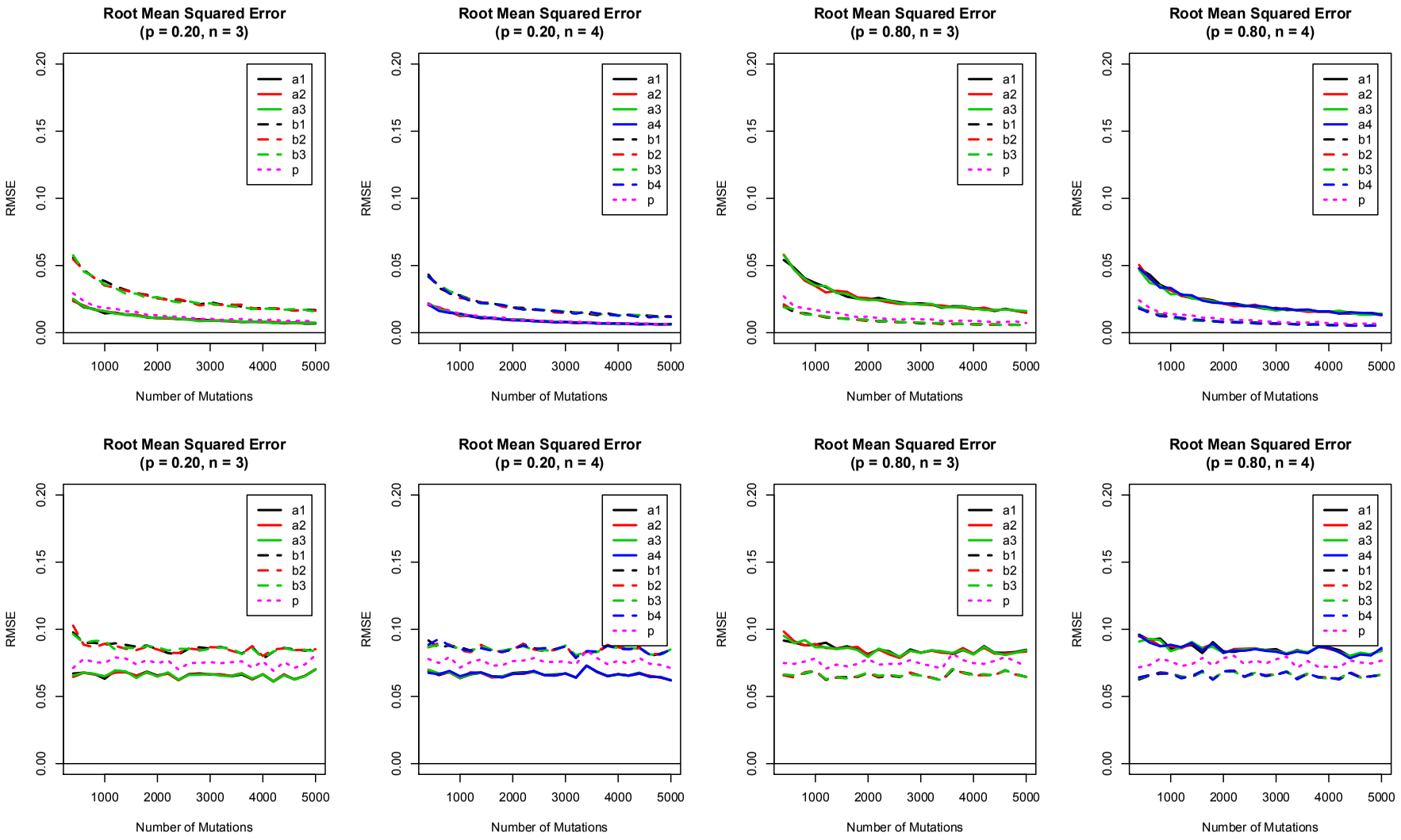

Figure 5.8 : Simulation Study 3: Varying n (Table 5.1). Assessing RMSE.
Row 1: Estimated with postMUT (simple), Row 2: Estimated with postMUT.

5.2 Performance on Simulated Data using the postMUT Model

Next, we simulate functional predictions using the postMUT model. Again, we consider three different simulation studies which are described in Table 5.2. We explore different ranges of the parameters similar to Section 5.1. We calculate d_j, e_j listed in Table 5.2 using the marginal probabilities in equations 4.4, so they match a_j and b_j in Table 5.1. Parameter estimation is performed using the postMUT (simple) and postMPUT models.

5.2.1 Simulation Study 1: Varying p

As we vary p, the bias and RMSE of d_j, e_j are very small when estimating the parameters using the postMUT model, but the bias and RMSE of a_j, b_j are much larger when estimating with the postMUT (simple) model and postMUT model. This is because the bias and RMSE of δ and γ are large (Figures 5.9, 5.10). Additionally, when p is small, $\hat{p}$ is positively biased, but negatively biased when p is large (Figure 5.9) when estimating with the postMUT (simple) and postMUT models.

5.2.2 Simulation Study 2: Varying a_j, b_j

Next, we compare bias and RMSE of algorithms with increasing in false positive rates and decreasing true positive rates (Models A, 2.1, 2.2 and 2.3). Similar to the results seen in Section 5.1, the RMSE(a_1) = RMSE(a_2) > RMSE(a_3) and RMSE(b_1) = RMSE(b_2) > RMSE(b_3) even though the sensitivity and specificity of algorithms 1 and 2 are better than algorithm 3 (Figures 5.11, 5.12). Again, this effect is not due to the model, but rather from the agreement seen in the disjoint categories (Figure 5.6).

Table 5.2 : Parameters used for simulation studies described Section 5.2

Simulation Simulation Studys		d_1	d_2	d_3	d_4	e_1	e_2	e_3	e_4	δ	γ	p
	Model A	0.056	0.056	0.056	-	0.994	0.994	0.994	-	0.90	0.90	0.20
Simulation Study 1: Varying p	Model A											
	Model 1.1	0.056	0.056	0.056	-	0.994	0.994	0.994	-	0.90	0.90	**0.05**
	Model 1.2	0.056	0.056	0.056	-	0.994	0.994	0.994	-	0.90	0.90	**0.80**
Simulation Study 2: Varying a_j, b_j	Model A											
	Model 2.1	0.056	0.056	**0.118**	-	0.994	0.994	**0.812**	-	0.90	0.90	0.20
	Model 2.2	0.056	**0.118**	**0.118**	-	0.994	**0.812**	**0.812**	-	0.90	0.90	0.20
	Model 2.3	**0.118**	**0.118**	**0.118**	-	**0.812**	**0.812**	**0.812**	-	0.90	0.90	0.20
Simulation Study 3: Varying n	Model A											
	Model 3.1	0.056	0.056	0.056	**0.056**	0.994	0.994	0.994	**0.994**	0.90	0.90	0.20
	Model 1.2											
	Model 3.2	0.056	0.056	0.056	**0.056**	0.994	0.994	0.994	**0.994**	0.90	0.90	0.80

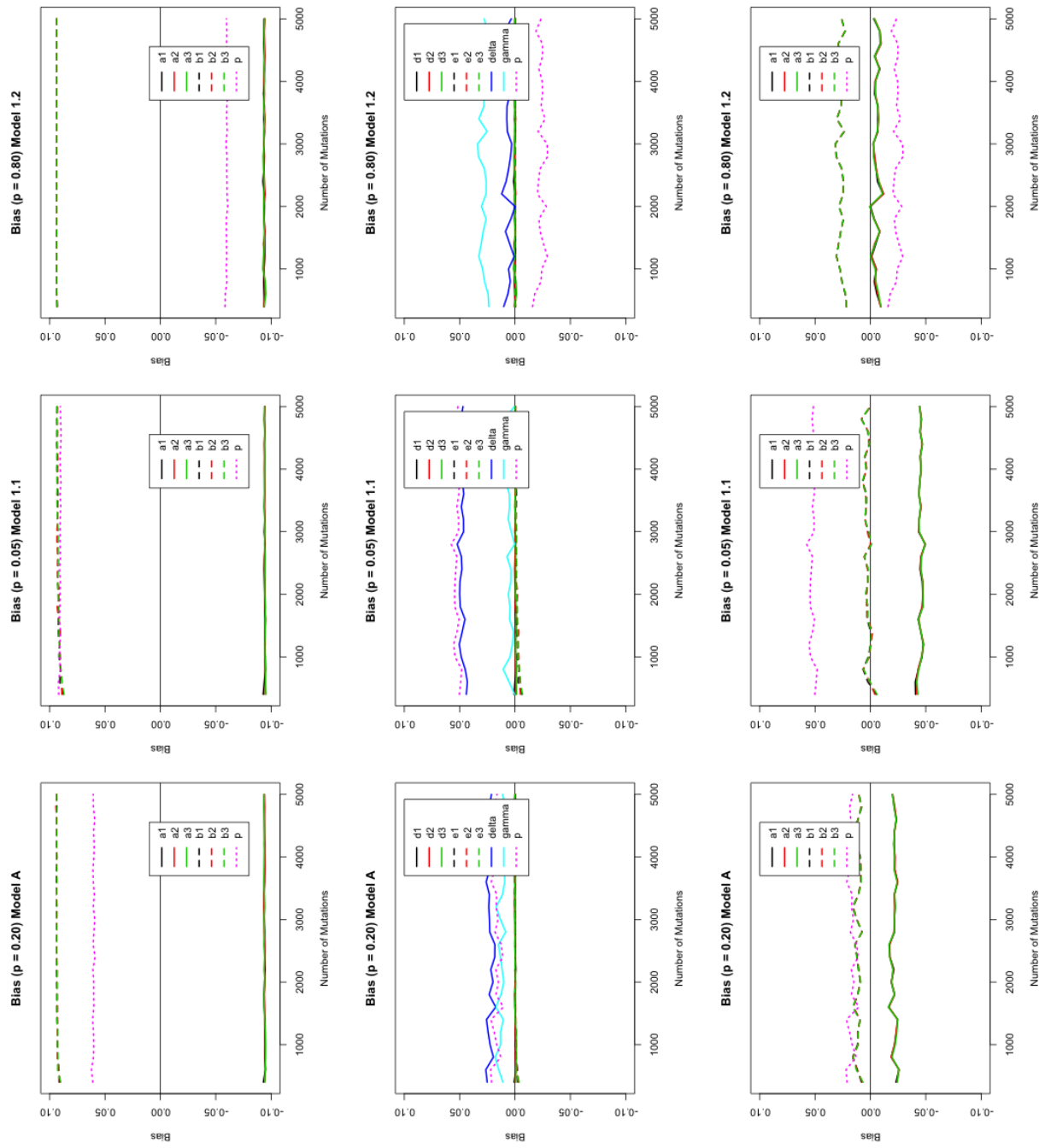

Figure 5.9 : Simulation Study 1: Varying p (Table 5.2). Assessing Bias using $p = 0.20$, $p = 0.05$ and $p = 0.80$. Row 1: Estimated with postMUT (simple), Rows 2 and 3: Estimated with postMUT.

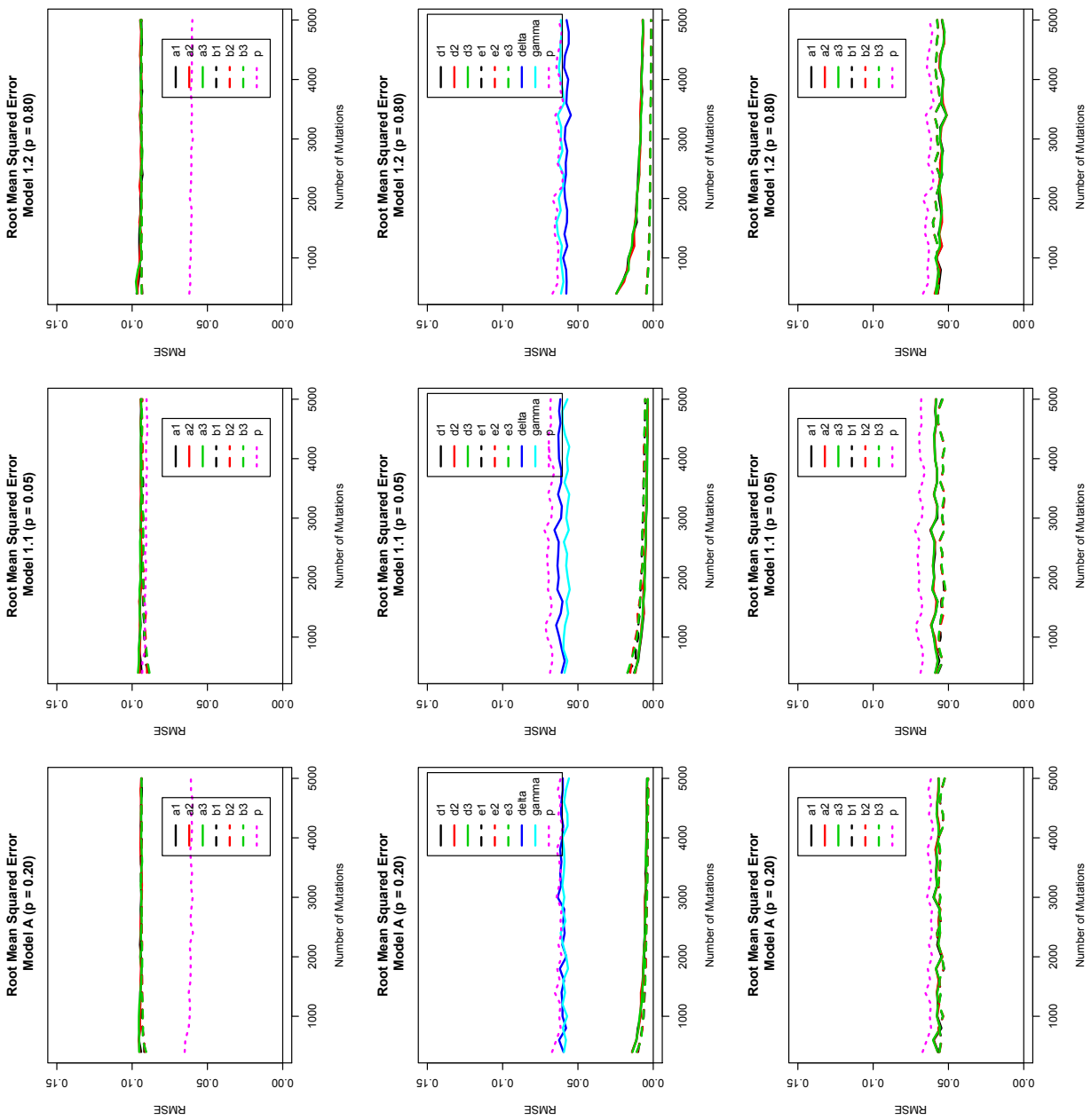

Figure 5.10 : Simulation Study 1: Varying p (Table 5.2). Assessing RMSE using $p = 0.20$, $p = 0.05$ and $p = 0.80$. Row 1: Estimated with postMUT (simple), Rows 2 and 3: Estimated with postMUT.

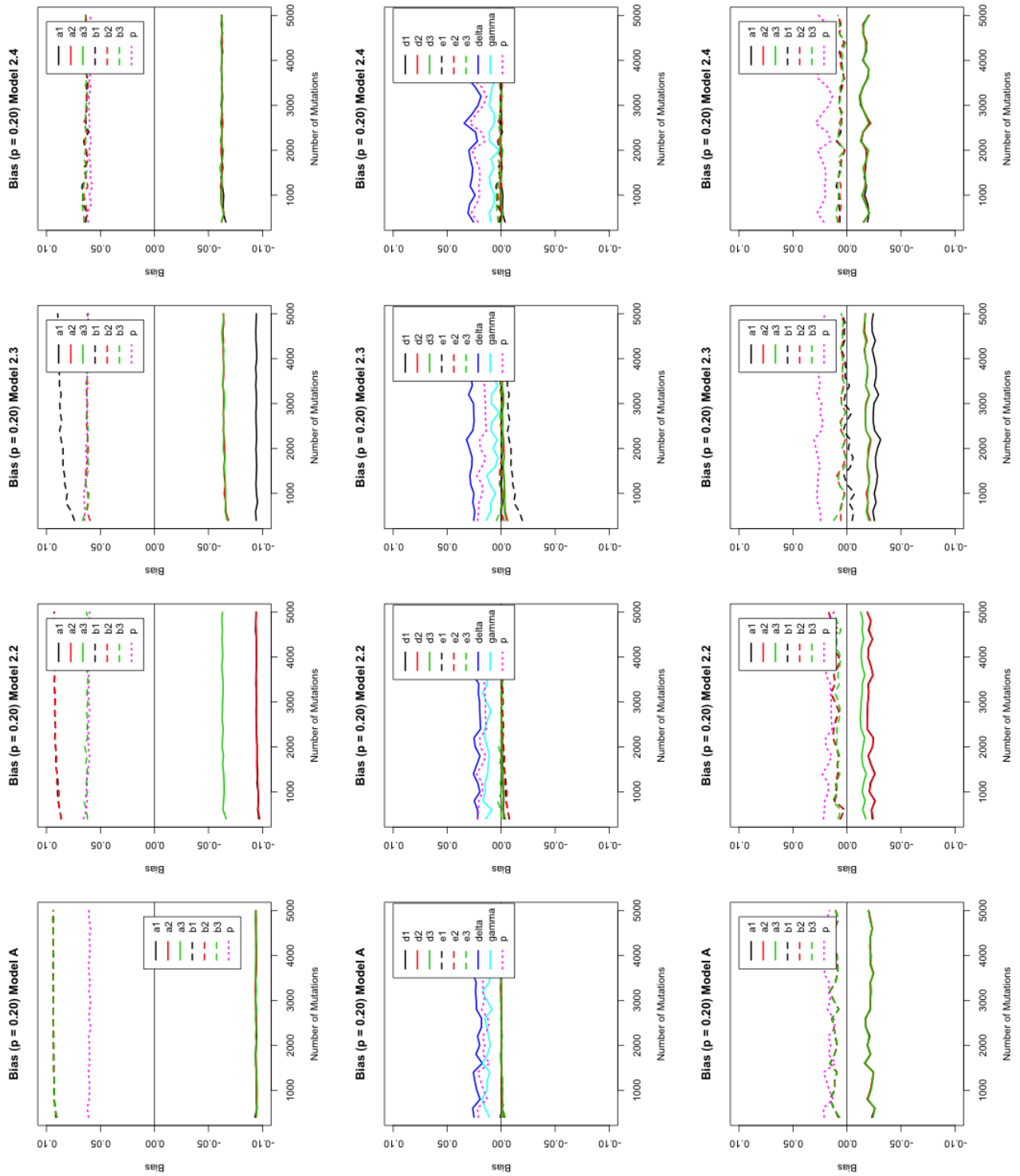

Figure 5.11 : Simulation Study 2: Varying a, b (Table 5.2). Assessing Bias.
Row 1: Estimated with postMUT (simple), Rows 2 and 3: Estimated with postMUT.

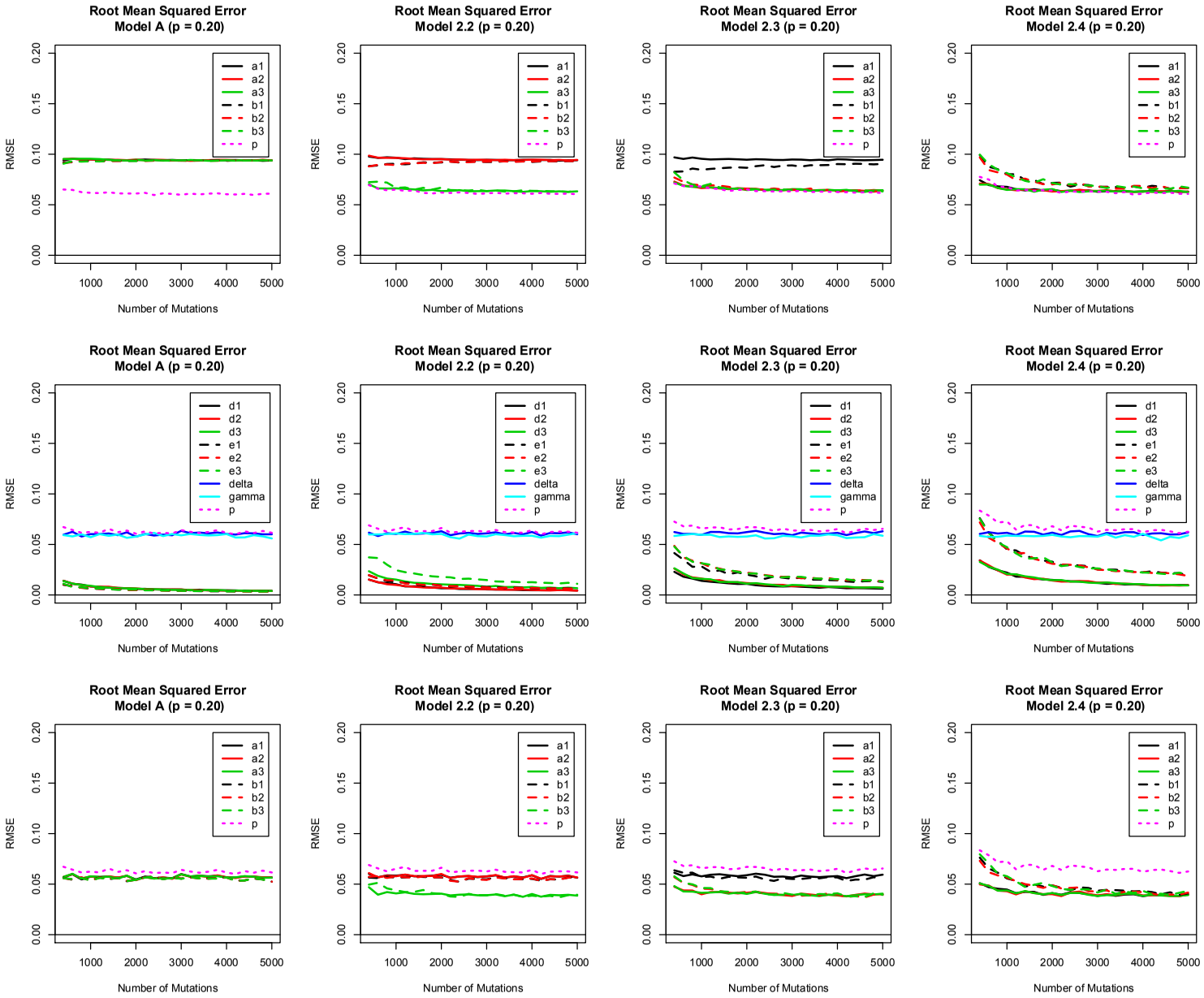

Figure 5.12 : Simulation Study 2: Varying a, b (Table 5.2). Assessing RMSE.
Row 1: Estimated with postMUT (simple), Rows 2 and 3: Estimated with postMUT.

5.2.3 Simulation Study 3: Varying n

When estimating using the postMUT model, the bias and RMSE of d_j and e_j decreases as the number of algorithms n increases. As noted in Simulation Study 1 because the bias and RMSE of δ and γ are large, we do not see a reduction in bias or RMSE when we calculate the marginal probabilities a_j and b_j (Figures 5.13, 5.14).

5.3 Discussion

In this chapter, we investigated the performance of the postMUT (simple) and postMUT models using simulation studies. We assessed the bias and RMSE for a statistical model we developed Chapter 4 based on the capture-recapture paradigm. These models combine discrete discordant functional predictions (neutral or deleterious) in a statistically rigorous manner and estimate a unified posterior probability for each mutation being deleterious. In absence of a gold standard (or set of mutations with known functionality), the models jointly estimate the sensitivities (probability of correctly predicting a deleterious mutation) and specificities (probability of correctly predicting a neutral mutation) of each *in silico* method and the overall proportion of deleterious mutations in the dataset using the Expectation-Maximization algorithm.

We previously gave examples Chapter 4 of how the two postMUT models perform on real sets of missense mutations with and without a 'gold standard' (mutations with known functionality). Even with assumptions of conditional independence between the algorithms, we showed the sensitivity and specificity estimates using the postMUT models closely match the sensitivity and specificity estimated directly using the known functional mutation status. We also discussed technical issues with the identifiability of the parameters when performing parameter estimation. These problems are well-

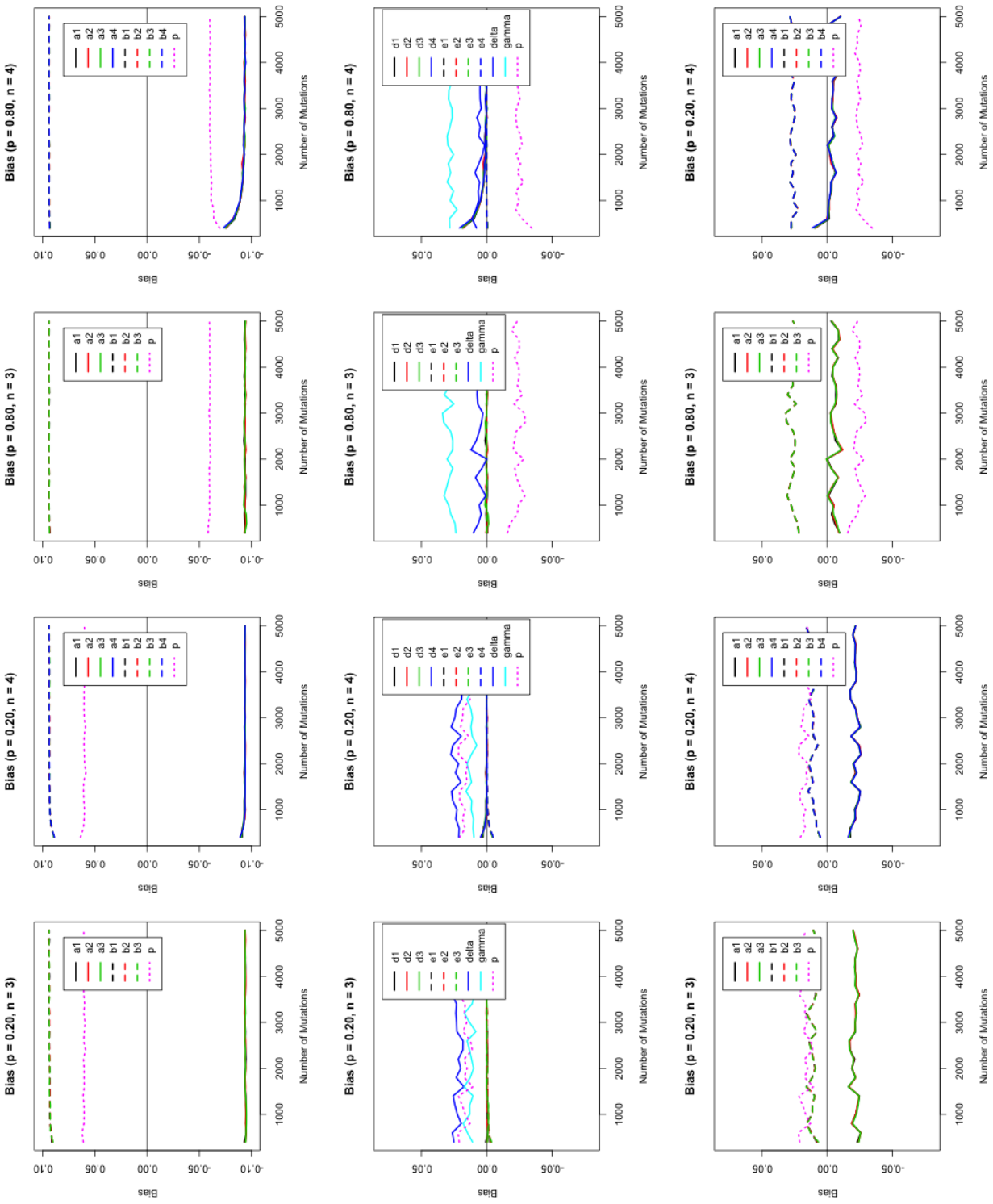

Figure 5.13 : Simulation Study 3: Varying n (Table 5.2). Assessing Bias.
Row 1: Estimated with postMUT (simple), Rows 2 and 3: Estimated with postMUT.

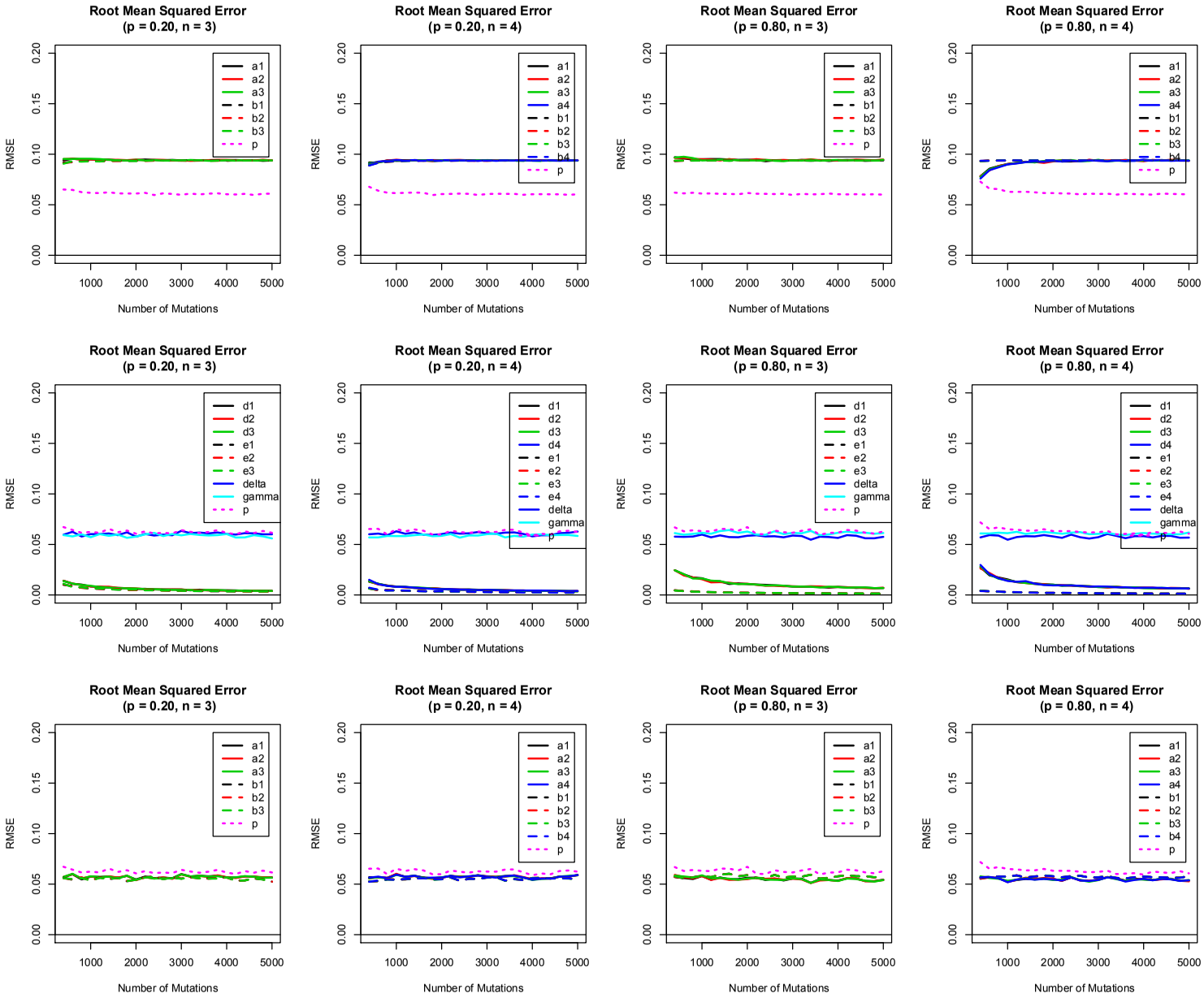

Figure 5.14 : Simulation Study 3: Varying n (Table 5.2). Assessing RMSE.
Row 1: Estimated with postMUT (simple), Rows 2 and 3: Estimated with postMUT.

recognized in a more general context of mixture models (McLahlan and Peel, 2000). A perfect algorithm would report $b_j = 1$ and $a_j = 0$, but in reality the true positive rate and false positive rate may each range from [0,1]. Therefore, we impose two identifiability constraints $a_j \in [0, 0.50]$ and $b_j \in [0.5, 1]$ when performing parameter estimation. If the parameters were outside this range, the algorithm would have worse accuracy than randomly flipping a coin. After performing the simulation studies, we feel improvements still need to be made to the postMUT model because we see the large bias and RMSE for the δ and γ parameters. One possible way to improve our model could be to incorporate additional information such as the continuous score or probability of a mutation being predicted deleterious (or neutral) produced from these algorithms to more accurately determine the true functionality of missense mutations.

Chapter 6

Identifying Regions of Identity-by-Descent

In this chapter, we develop several hidden Markov models (HMMs) to identify regions of *identity-by-descent* (IBD) (or chromosomal segments in related individuals inherited from the same haplotype from each parent) using the observed *identity-by-state* (IBS) status from whole-exome sequencing data. These HMMs use the IBS status which describes the number of shared alleles between two affected siblings. In consanguineous families, the shared alleles will be homozygous, but in non-consanguineous families, the shared alleles may be heterozygous, but identically shared in each sibling.

We consider several extensions of a previously developed first-order HMM [Rödelsperger et al., 2011] by exploring conditional emission probabilities and also a second-order dependence structure between the observed variant calls in siblings. Due to the structure of the conditional emission probabilities we derive a a Viterbi-type algorithm to predict regions of IBD. These models are inhomogeneous because the transition probabilities at each position vary depending on the position and sex-specific recombination rates. In addition, we show the emission probabilities used in the HMMs are a function of the minor allele frequency suggesting minor allele frequency should be used in determining IBD regions using whole-exome sequencing data.

In Section 6.1, we discuss the whole-exome sequencing data used in this chapter. Section 6.1.1 describes the procedure for simulating whole-exome data and Section 6.1.2 introduces a set of whole-exome sequencing data from a family affected by acute

lymphocytic leukemia and lymphoma. Table 6.1 describes five HMMs which we refer to as the Rodelsperger et al. (2011) Model, Model A (Section 6.3), Model B (Section 6.4), Models C and D (Section 6.5). In Section 6.3, we derive a first-order HMM with conditional emission probabilities which depend not only on the IBD status, but also the IBS status at the previous position. In this section, we treat IBD as a binary status of either IBD $\neq 2$ or IBD $= 2$. In Section 6.4, we derive a second-order HMM with second-order transition probabilities and and emission probabilities. Then in Section 6.5, we re-define the IBD and IBS status as $\{0, 1, 2\}$ (representing the number of alleles shared between pairs of individuals) and use a first-order HMM with conditional emissions.

We compare the results from the HMMs using simulated exome sequencing data and real exome sequencing data from two affected siblings in an autosomal dominant family discussed in Section 6.1.2 as an analysis in Section 6.7.

6.1 Exome Data

6.1.1 Simulated Exome Data

Following Rodelsperger et al. (2011), I simulated $N = 1000$ families with $n = 2$ siblings. First, I downloaded the genotypes of trios from Hapmap (Phase 3, hg18) [International HapMap Consortium, 2003] and extracted all female and male founders from the population. Parents were randomly sampled (1 maternal and 1 paternal) by sampling diploid chromosomes from individuals. To simulate the offspring, I generate 1 recombinant gamete from a female and 1 recombinant gamete from a male to make 1 offspring (repeat this process for n siblings). Next, a subset of the genotypes are kept by only using the positions in the CCDS coordinates [Pruitt et al., 2009].

For each sibling, I simulated genotype calling errors with probability $\epsilon = 0.05$ and sequencing false-positive calls with probability $\delta = 0.001$. Finally, positions were eliminated where there were no variant calls in n siblings. If there was at least one variant observed in one of the siblings, then the IBD and IBS status of this variant was recorded. For families with two siblings the expected proportion of the genome that is IBD = 2 is $\frac{1}{4}$. Figure 6.1 shows the proportion of simulated genomes that are IBD = 2 matches the expected number.

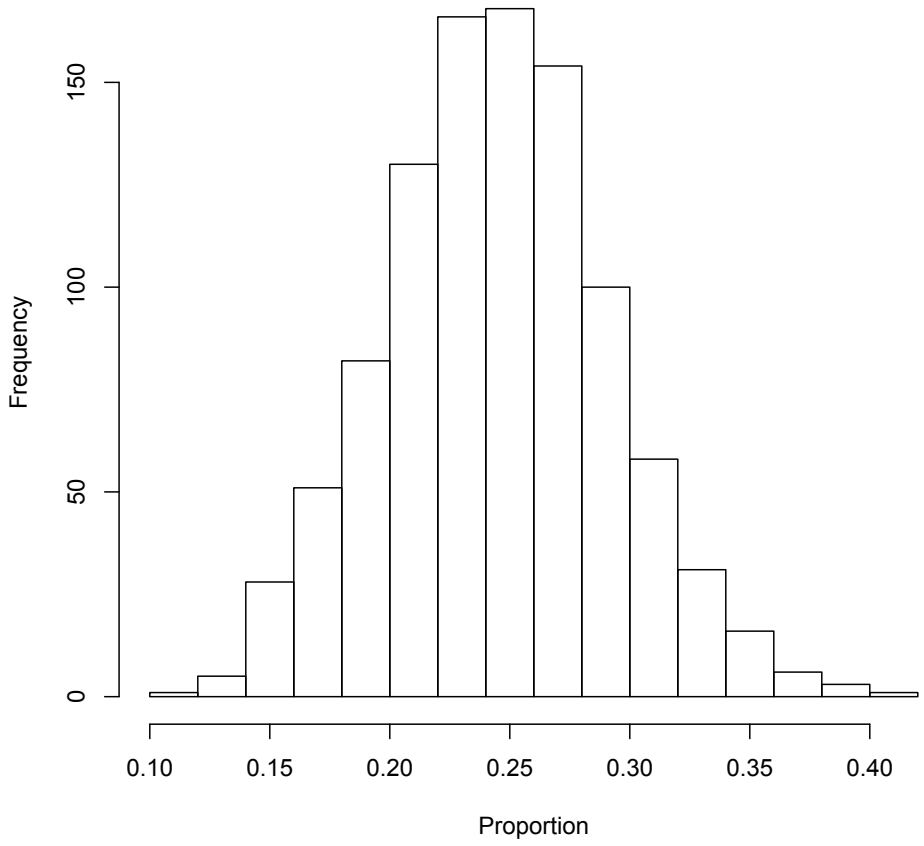

Figure 6.1 : Proportion of simulated exomes that are IBD=2.

6.1.2 Whole-Exome Sequencing Data

In addition to simulated whole-exome sequencing data, we use the unpublished real human exome sequencing data discussed in Section 4.3.5. The whole exome sequencing data from two siblings (FCP502 and FCP416) were used in this analysis. We note that this family does not follow a recessive inheritance pattern which is the inheritance pattern that the HMM in this thesis was developed for. This family with an autosomal dominant inheritance pattern is used as just an example.

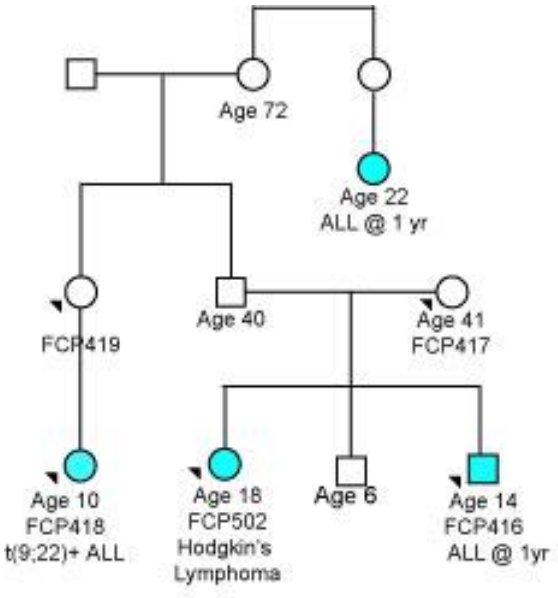

Figure 6.2 : Extended Pedigree with four cases of lymphocytic leukemia and lymphoma. Same pedigree considered as in Section 4.3.5.

6.2 Inhomogeneous HMMs for predicting regions of IBD

In this section, several extensions of a first-order HMM [Rödelsperger et al., 2011] previously developed are considered. The models considered in this chapter are listed

in in Table 6.1.

The first-order HMM developed by Rodelsperger et al. (2011) is listed first. Model A is defined as 1st-order HMM, but with conditional emission probabilities, $P[X_n = b|X_{n-1} = c, \pi_n = i]$. These emission probabilities depend not only on the unobserved IBD status at position n, but also the observed IBS status at position $n-1$. We define a new Viterbi-type algorithm to predict regions of IBD with these conditional emission probabilities. Model B extends the first-order HMM to a second-order HMM by investigating the second-order dependence structure between the observed variant calls in siblings. The emission probabilities are also second-order, $P[X_n = b|\pi_n = j, \pi_{n-1} = i]$. Model C redefines the IBD status from IBD = 2 or IBD $\neq$ 2 to IBD = 0, 1, 2. It also redefines the IBS status to IBS = 0, 1, 2 (i.e. number of shared alleles between two affected siblings). Model D is estimated using Model C, but after obtaining the predicted IBD regions $\in \{0, 1, 2\}$ we convert the IBD status to IBD = 2, IBD $\neq$ 2 to compare to first three models. Finally, we consider emission probabilities as a function of minor allele frequency.

6.3 Model A

In this section we consider a first-order inhomogenous HMM but with conditional emission probabilities. Let π_n be a first-order inhomogeneous Markov process where

$$\pi_n = \begin{cases} 1 & \text{if position } n \text{ is IBD} = 2 \text{ in two siblings} \\ 0 & \text{otherwise} \end{cases}$$

The IBD status π_n is not observable, but we do observe the IBS status X_n

$$X_n = \begin{cases} 1 & \text{if position } n \text{ is IBS}^* \text{ in two siblings} \\ 0 & \text{otherwise} \end{cases}$$

Table 6.1 : A list of the inhomogeneous HMMs considered in Chapter 6

Model	IBD status	IBS status	Order	Transition probabilities	Conditional emissions?	Emission probabilities
Rodelsperger et al. (2011)	IBD $= 2, \neq 2$	IBS $= 2, \neq 2$	1st	$P[\pi_{n+1} = j \mid \pi_n = i]$	No	$P[X_n = b \mid \pi_n = i]$
Model A	IBD $= 2, \neq 2$	IBS $= 2, \neq 2$	1st	$P[\pi_{n+1} = j \mid \pi_n = i]$	Yes	$P[X_n = b \mid X_{n-1} = c, \pi_n = i]$
Model B	IBD $= 2, \neq 2$	IBS $= 2, \neq 2$	2nd	$P[\pi_{n+1} = k \mid \pi_n = j, \pi_{n-1} = i]$	No	$P[X_n = b \mid \pi_n = j, \pi_{n-1} = i]$
Model C	IBD $= 0, 1, 2$	IBS $= 0, 1, 2$	1st	$P[\pi_{n+1} = j \mid \pi_n = i]$	Yes	$P[X_n = b \mid X_{n-1} = c, \pi_n = i]$
Model D	Predict IBD (0, 1, 2) regions with Model C. Convert predictions to IBD $= 2, \neq 2$ and compare to Models A, B.					

where IBS* is defined as each sibling being called to the same homozygous or heterozygous genotype.

6.3.1 First-order Transition Probabilities

The probability that position n is IBD = 2 in two siblings is

$$P_1 = P[\pi_n = 1] = (\frac{1}{4})^2$$

and the probability that position n is IBD $\neq$ 2 is

$$P_0 = P[\pi_n = 0] = 1 - (\frac{1}{4})^2$$

For any two loci (n and $n+1$), the first-order inhomogeneous Markov chain is characterized by the transition probability matrix

$$P_{ij} = P[\pi_{n+1} = j | \pi_n = i]$$

where the transition probabilities depend on the sex-specific recombination rates between n and $n+1$ loci positions for the gender s. The recombination rates are defined as $\theta_{n,n+1,s}$. These rates can be easily obtained from the UCSC Browser with the rtracklayer R package [Lawrence et al., 2009].

To compute these transition probabilities, the conditional probability that given loci n is IBD = 2, the probability that loci $n+1$ is IBD = 2 in two siblings is given by

$$P_{11} = [(1 - \theta_{n,n+1,p})^2 + (\theta_{n,n+1,p})^2][(1 - \theta_{n,n+1,m})^2 + (\theta_{n,n+1,m})^2]$$

Then the joint probability that a pair of adjoining loci (n, $n+1$) are both IBD = 2:

$$P[\pi_n = 1, \pi_{n+1} = 1] \quad = \quad P[\pi_{n+1} = 1 | \pi_n = 1] P[\pi_n = 1] = P_{11} P_1$$

This leads to the following conditional and joint probabilities:

$$\begin{aligned}
P_{01} &= P[\pi_{n+1}=1|\pi_n=0] = \frac{P[\pi_n=0|\pi_{n+1}=1]P[\pi_{n+1}=1]}{P[\pi_n=0]} \\
&= \frac{(1-P[\pi_n=1|\pi_{n+1}=1])P[\pi_{n+1}=1]}{P[\pi_n=0]} \\
&= \frac{(1-\frac{P[\pi_n=1,\pi_{n+1}=1]}{P[\pi_{n+1}=1]})P[\pi_{n+1}=1]}{P[\pi_n=0]} \\
&= \frac{P[\pi_{n+1}=1]-P[\pi_n=1,\pi_{n+1}=1]}{P[\pi_n=0]} = \frac{P_1-P_{11}}{P_0} \\
P_{10} &= 1-P_{11} \\
P_{00} &= 1-P_{01}
\end{aligned}$$

6.3.2 First-order Conditional Emissions

The conditional emission probabilities are given by

$$e_{(X_{n-1},\pi_n)}(X_n) = P[X_n|X_{n-1},\pi_n]$$

We estimate the conditional emission probabilities two ways: 1) using the empirical frequency estimates from the simulated families (Section 6.1.1) and 2) simulating 3-loci and estimating the conditional emissions as a function of minor allele frequency. Several groups using SNP array data have used minor allele frequencies and haplotype frequencies as a way to estimate emission probabilities [Albrechtsen et al., 2009, Han and Abney, 2011], but have not tried to assess their influence using whole-exome sequencing data. To do this, we simulated data with three points representing the three loci $n-2$, $n-1$, and n in the following procedure:

1. Repeat the following 10 times:

   (a) Repeat the following 1,000,000 times:

i. For 3 positions, define MAFs for position $n-1$ and n: $p(a)$, $p(b)$, $p(c)$. Compute genotype probabilities: $p(AA)$, $p(Aa)$, $p(aa)$, $p(BB)$, $p(Bb)$, $p(bb)$, $p(CC)$, $p(Cc)$, $p(cc)$.

ii. Simulate mother and father genotypes at 3 positions. Convert to haplotypes.

iii. Simulate maternal and paternal haplotypes for each sibling using defined recombination frequency between position $n-2$ and $n-1$ and recombination frequency between position $n-1$ and n. Simulate genotyping error with probability $\epsilon = 0.05$. Determine IBS and IBD status.

iv. Record what event was simulated (e.g. $X_n = 1$, $\pi_n = 2$)

(b) Calculate emission probabilities:

i. Rodelsperger 1st-order emissions: $P[X_n = 1|\pi_n = 1]$

ii. 1st-order conditional emissions: $P[X_n = 1|X_{n-1}, \pi_n = 1]$

2. Calculate mean and standard deviation of emission probabilities

We used the 1000 Genomes minor allele frequencies (Feb 2012 release) for each set of mutations using wANNOVAR [Wang et al., 2010, Chang and Wang, 2012]

As show in Figure 6.3.2, the emission probabilities conditional on IBD $\neq 2$ at position n from the Rodelsperger et al. (2011) Model vary as function of minor allele frequency. Figure 6.3.2 shows the emission probabilities in Model A also vary as a function of minor allele frequency.

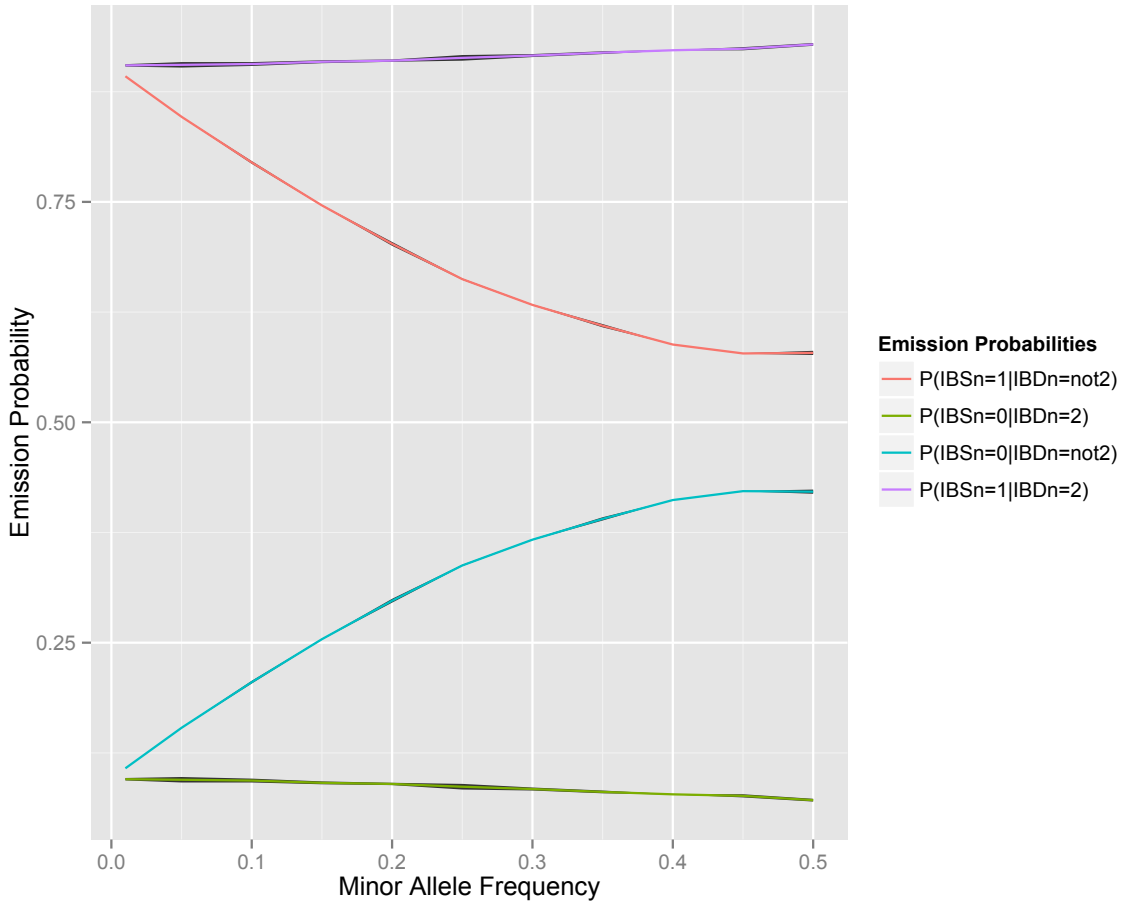

Figure 6.3 : Emission probabilities in Rodelsperger et al. (2011) model as a function of MAF

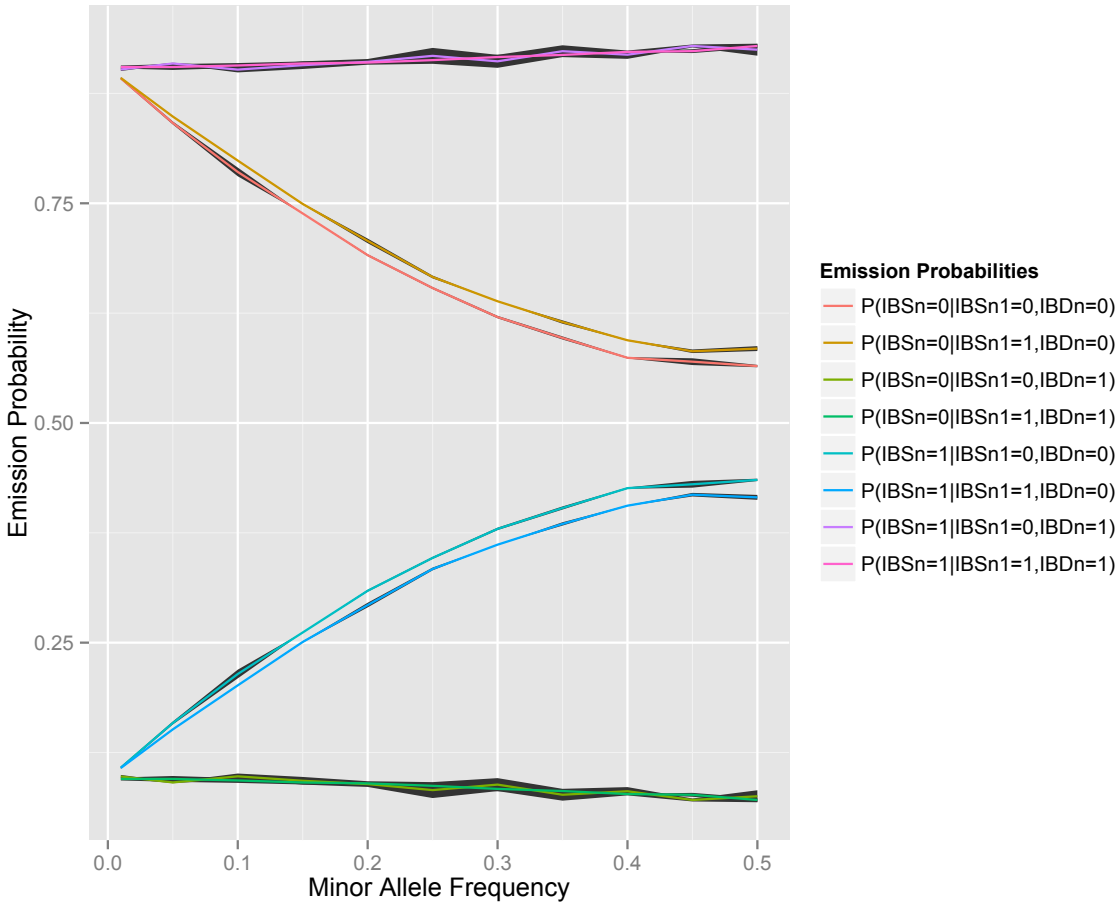

Figure 6.4 : Emission probabilities in Model A as a function of MAF

6.3.3 New Joint Distribution (X, π)

Because the emissions are conditional on both the IBD and IBS status, we derived the new joint distribution of (X, π). Consider

$$\begin{aligned} E_n &= P[X_1, \ldots, X_n | \pi_1, \ldots, \pi_n] \\ &= P[X_n | X_1, \ldots, X_{n-1}, \pi_1, \ldots, \pi_n] P[X_1, \ldots, X_{n-1} | \pi_1, \ldots, \pi_n] \\ &= P[X_n | X_{n-1}, \pi_n] E_{n-1} \end{aligned}$$

This leads to the following relationship for the entire sequence of length L

$$E_L = P[X|\pi] = E_1 \prod_{n=1}^{L} P[X_n | X_{n-1}, \pi_n]$$

Next, consider the joint distribution of π up until the nth position

$$\begin{aligned} \Pi_n &= P[\pi_1, \ldots, \pi_n] \\ &= P[\pi_n | \pi_1, \ldots, \pi_{n-1}] P[\pi_1, \ldots, \pi_{n-1}] \\ &= P[\pi_n | \pi_{n-1}] \Pi_{n-1} \end{aligned}$$

Then,

$$\Pi_L = P[\pi] = \Pi_1 \prod_{n=1}^{L} P[\pi_n | \pi_{n-1}]$$

Then, the joint distribution of X and π is given by

$$\begin{aligned} P(X, \pi) &= P[X|\pi] P[\pi] = E_L \Pi_L \\ &= E_1 \Pi_1 \prod_{n=1}^{L} P[X_n | X_{n-1}, \pi_n] P[\pi_n | \pi_{n-1}] \\ &= E_1 \Pi_1 \prod_{n=1}^{L} e_{(X_{n-1}, \pi_n)}(X_n) P_{\pi_n, \pi_{n-1}} \end{aligned}$$

where $e_{(X_{n-1}, \pi_n)}(X_n)$ is a conditional emission probability for the nth position and $P_{\pi_n, \pi_{n-1}}$ is the transition probability at the nth step given the previous step.

6.3.4 New Iterative Viterbi-type Algorithm

To make inference on the hidden path π, we may define a Viterbi-type algorithm that incorporates the new conditional emission probabilities. We use an iterative Forward Algorithm that maximizes the likelihood $P(X, \pi)$ with respect to the unknown parameters $\theta = (e, P)$. Consider

$$\begin{aligned}
\max_{\pi} P(X,\pi) &= \max_{\pi_1,\ldots,\pi_L} E_1\Pi_1 \prod_{n=1}^{L} P[X_n|X_{n-1},\pi_n]P[\pi_n|\pi_{n-1}] \\
&= \max_{\pi_1,\ldots,\pi_L} P[X_L|X_{L-1},\pi_L]P[\pi_L|\pi_{L-1}] \times P[X_{L-1}|X_{L-2},\pi_{L-1}]P[\pi_{L-1}|\pi_{L-2}] \times \ldots \\
&\quad \ldots \times P[X_3|X_2,\pi_3]P[\pi_3|\pi_2] \times P[X_2|X_1,\pi_2]P[\pi_2|\pi_1] \times P[X_1|\pi_1]P[\pi_1|\pi_0]P[\pi_0] \\
&= \max_{\pi_L}\{\ P[X_L|X_{L-1},\pi_L] \max_{\pi_1,\ldots,\pi_{L-1}}\{\ P[\pi_L|\pi_{L-1}] \times P[X_{L-1}|X_{L-2},\pi_{L-1}] \\
&\quad \times P[\pi_{L-1}|\pi_{L-2}] \times \ldots \times P[X_2|X_1,\pi_2]P[\pi_2|\pi_1] \times P[X_1|\pi_1]P[\pi_1|\pi_0]P[\pi_0] \quad \}\ldots\} \\
&= \max_{\pi_L}\{\ P[X_L|X_{L-1},\pi_L] \max_{\pi_{L-1}}\{\ P[\pi_L|\pi_{L-1}] \times P[X_{L-1}|X_{L-2},\pi_{L-1}] \\
&\quad \max_{\pi_{L-2}}\{\ P[\pi_{L-1}|\pi_{L-2}] \times \ldots \times P[X_4|X_3,\pi_4] \max_{\pi_3}\{\ P[\pi_4|\pi_3] \times \\
&\quad \times P[X_3|X_2,\pi_3] \max_{\pi_2}\{\ P[\pi_3|\pi_2] \times P[X_2|X_1,\pi_2] \\
&\quad \max_{\pi_1}\{\ P[\pi_2|\pi_1] \times P[X_1|\pi_1] \max_{\pi_0}\{\ P[\pi_1|\pi_0] \times P[\pi_0]\ \}\ldots\}
\end{aligned}$$

Implementing New Iterative Viterbi-type Algorithm

The inhomogeneous transition probabilities and emission probabilities for Model A discussed in this section were programmed in R. We implemented the iterative Viterbi-type algorithm to calculate the path or predict the IBD states.

6.4 Model B

In this section, we extend the first-order inhomogenous HMM [Rödelsperger et al., 2011] to a second-order HMM to investigate the second-order dependence structure between the observed variant calls in siblings. Let π_n be a second-order inhomogeneous Markov process where

$$\pi_n = \begin{cases} 1 & \text{if position } n \text{ is IBD} = 2 \text{ in two siblings} \\ 0 & \text{otherwise} \end{cases}$$

The IBD status π_n is not observable, but we do observe the IBS status X_n

$$X_n = \begin{cases} 1 & \text{if position } n \text{ is IBS}^* \text{ in two siblings} \\ 0 & \text{otherwise} \end{cases}$$

where IBS* is defined as each sibling being called to the same homozygous or heterozygous genotype.

6.4.1 Second-order Transition Probabilities

In Section 6.3.1, we described the first-order transition probabilities using any two loci(n, $n+1$). For any three loci $(n-1,\ n,\ n+1)$, the second-order inhomogeneous Markov process is characterized the transition probability matrix

$$P_{ijk} = P[\pi_{n+1} = k | \pi_n = j, \pi_{n-1} = i]$$

The second-order transition probabilities depend on the sex-specific recombination rates between the $n-1$ and n loci positions and between the n and $n+1$ loci positions for the gender s. The recombination rates are defined as

$$\theta_{n-1,n,s} = \theta(n-1, n, s) \qquad \text{and} \qquad \theta_{n,n+1,s} = \theta(n, n+1, s)$$

These rates can be easily obtained from the UCSC Browser with the rtracklayer R package [Lawrence et al., 2009].

Using these recombination rates, we derive the second-order transition probabilities. The conditional probability that given loci $n-1$ and n is IBD $= 2$, the probability that loci $n+1$ is IBD $= 2$ in two siblings is given by

$$P_{111} = \frac{P[\pi_{n+1} = 1, \pi_n = 1, \pi_{n-1} = 1]}{P[\pi_n = 1, \pi_{n-1} = 1]}$$

which equals

$$\{\ [((1-\theta_{n-1,n,p})(1-\theta_{n,n+1,p}))^2 + ((\theta_{n-1,n,p})(1-\theta_{n,n+1,p}))^2 + ((1-\theta_{n-1,n,p})(\theta_{n,n+1,p}))^2 + ((\theta_{n-1,n,p})(\theta_{n,n+1,p}))^2]$$
$$\times\ [((1-\theta_{n-1,n,m})(1-\theta_{n,n+1,m}))^2 + ((\theta_{n-1,n,m})(1-\theta_{n,n+1,m}))^2 + ((1-\theta_{n-1,n,m})(\theta_{n,n+1,m}))^2 + ((\theta_{n-1,n,m})(\theta_{n,n+1,m}))^2]$$
$$\times\ P_1\ \}/P[\pi_n = 1, \pi_{n-1} = 1]$$

Then the joint distribution for the three loci $(n-1, n, n+1)$ being IBD $= 2$ is

$$P[\pi_{n+1} = 1, \pi_n = 1, \pi_{n-1} = 1] = P_{111}P_{11}$$

Similarly,

$$\begin{aligned}
P_{011} &= P[\pi_{n+1} = 1 | \pi_n = 1, \pi_{n-1} = 0] \\
&= \frac{P[\pi_{n-1} = 0 | \pi_n = 1, \pi_{n+1} = 1] P[\pi_{n+1} = 1, \pi_n = 1]}{P[\pi_n = 1, \pi_{n-1} = 0]} \\
&= \frac{(1 - P[\pi_{n-1} = 1 | \pi_n = 1, \pi_{n+1} = 1]) P[\pi_{n+1} = 1, \pi_n = 1]}{P[\pi_n = 1, \pi_{n-1} = 0]} \\
&= \frac{(1 - \frac{P[\pi_{n+1}=1, \pi_n=1, \pi_{n-1}=1]}{P[\pi_{n+1}=1, \pi_n=1]}) P[\pi_{n+1} = 1, \pi_n = 1]}{P[\pi_n = 1, \pi_{n-1} = 0]} \\
&= \frac{P[\pi_n = 1, \pi_{n+1} = 1] - P[\pi_{n-1} = 1, \pi_n = 1, \pi_{n+1} = 1]}{P[\pi_n = 0, \pi_{n-1} = 0]} \\
&= \frac{P_{11} - P_{111}}{P_{00}} \\
P_{110} &= 1 - P_{111} \\
P_{010} &= 1 - P_{011}
\end{aligned}$$

and

$$\begin{aligned} P[\pi_{n+1}=1,\pi_n=1,\pi_{n-1}=0] &= P_{011}P_{01} \\ P[\pi_{n+1}=0,\pi_n=1,\pi_{n-1}=1] &= P_{110}P_{11} \\ P[\pi_{n+1}=0,\pi_n=1,\pi_{n-1}=0] &= P_{010}P_{01} \end{aligned}$$

These conditional and joint probabilities are only for half of the transition probability matrix. To fill in the other half, we consider the following scenario. Consider two siblings and P_{101} be the probability that both siblings are IBD = 2 at loci $n-1$ and $n+1$, but not at loci n. The only way for this to occur is if one of the siblings had a recombination between loci $(n-1,n)$ and $(n,n+1)$. Thus, if we consider two siblings then P_{101} translates to <u>one</u> sibling (but not both) had a recombination between [loci $(n-1,n)$ and $(n,n+1)$. Hence, we define P_{101} as

$$[[\theta_{n-1,n,p}\theta_{n,n+1,p}]\{[(1-\theta_{n-1,n,p})(1-\theta_{n,n+1,p})]+[\theta_{n-1,n,p}(1-\theta_{n,n+1,p})]+[(1-\theta_{1,2,p})\theta_{2,3,p}]\}]$$
$$\times\ [[\theta_{n-1,n,m}\theta_{n,n+1,m}]\{[(1-\theta_{n-1,n,m})(1-\theta_{n,n+1,m})]+[\theta_{n-1,n,m}(1-\theta_{n,n+1,m})]+[(1-\theta_{n-1,n,m})\theta_{n,n+1,m}]\}]$$

Thus, the joint distribution is given by

$$P[\pi_{n+1}=1,\pi_n=0,\pi_{n-1}=1]=P_{101}P_{10}$$

Similarly,

$$\begin{aligned} P_{001} &= \frac{P[\pi_{n-1}=0|\pi_n=0,\pi_{n+1}=1]P[\pi_n=0,\pi_{n+1}=1]}{P[\pi_n=0,\pi_{n-1}=0]} \\ &= \frac{(1-\frac{P[\pi_{n-1}=1,\pi_n=0,\pi_{n+1}=1]}{P[\pi_n=0,\pi_{n+1}=1]})P[\pi_n=0,\pi_{n+1}=1]}{P[\pi_n=0,\pi_{n-1}=0]} \\ &= \frac{P_{01}-P_{101}}{P_{00}} \\ P_{100} &= 1-P_{101} \\ P_{000} &= 1-P_{001} \end{aligned}$$

and

$$\begin{aligned} P[\pi_{n+1}=1,\pi_n=0,\pi_{n-1}=0] &= P_{001}P_{00} \\ P[\pi_{n+1}=0,\pi_n=1,\pi_{n-1}=0] &= P_{010}P_{01} \\ P[\pi_{n+1}=0,\pi_n=0,\pi_{n-1}=0] &= P_{000}P_{00} \end{aligned}$$

6.4.2 Second-order Emission Probabilities

Our second-order extension of Rodelsperger et al. (2011) emissions probabilities are given by

$$e_{ij}(b) = P[X_n = b|\pi_n = j, \pi_{n-1} = i]$$

To estimate these emission probabilities, we used the same procedure as in Section 6.3.2. The emission probabilities which were estimated using the empirical frequencies using the simulated families were programmed. In addition, the emission probabilities estimated as a function of MAF at loci $n-2$, $n-1$ and n were also considered. As show in Figure 6.4.2, the emission probabilities conditional on IBD $\neq 2$ at position n from Model B vary as function of minor allele frequency.

6.4.3 Second-order Viterbi Algorithm

The inhomogeneous transition probabilities and emission probabilities discussed in this section were programmed in R. We used the 1000 Genomes minor allele frequencies (Feb 2012 release) for each set of mutations using wANNOVAR [Wang et al., 2010, Chang and Wang, 2012] for the emission probabilities. A second-order Viterbi algorithm [Thede and Harper, 1999] was employed to predict the loci IBD = 2 or IBD $\neq$ 2. A marginal posterior probability of being in the IBD = 2 state was also estimated.

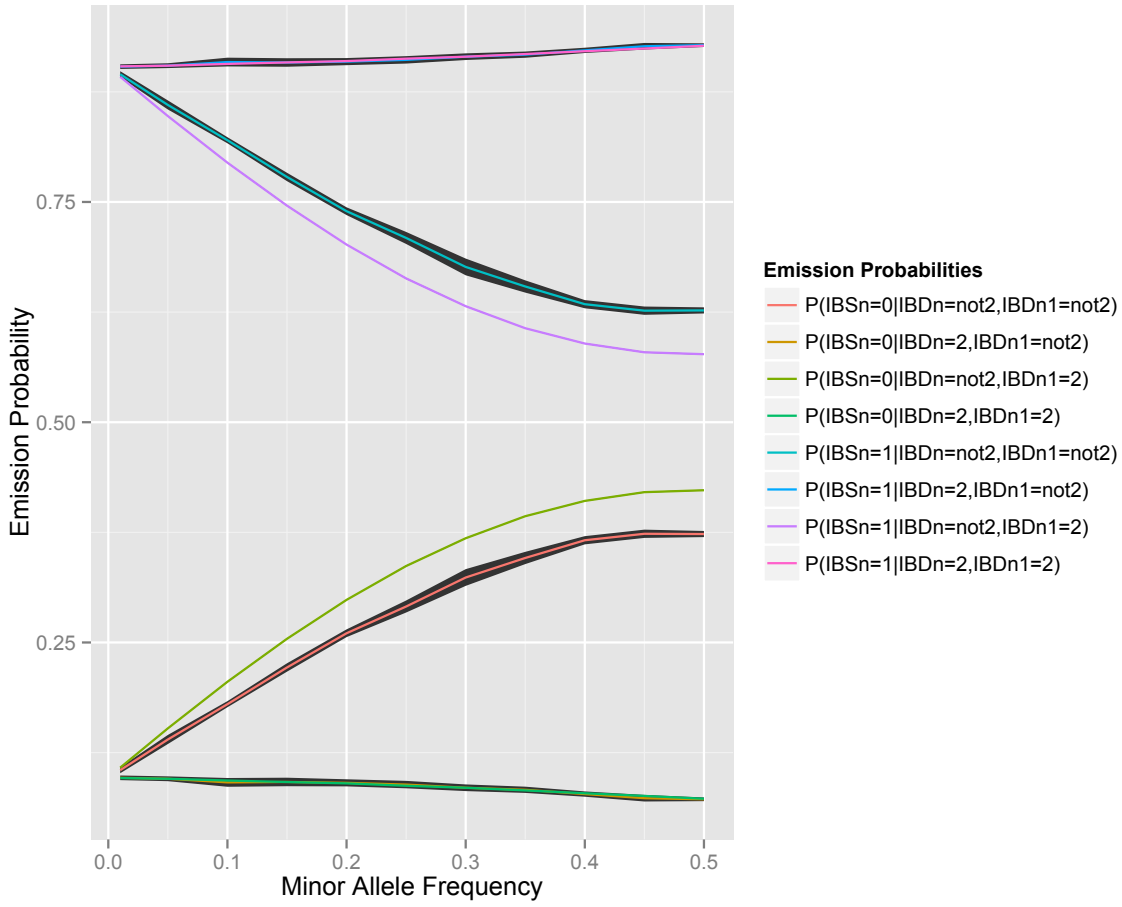

Figure 6.5 : Emission probabilities in Model B as a function of MAF

6.5 Models C and D

For this last model, we considered an inhomogenous first-order HMM as in Section 6.3 (Model A), but we redefine the IBD status as an ordinal variable $\in \{0, 1, 2\}$ where the status represents the number of alleles shared IBD between pairs of individuals. We also redefined the IBS status $\in \{0, 1, 2\}$.

Let π_n by a first-order inhomogeneous Markov process where

$$\pi_n = \begin{cases} 2 & \text{if position } n \text{ is IBD} = 2 \text{ in two siblings} \\ 1 & \text{if position } n \text{ is IBD} = 1 \text{ in two siblings} \\ 0 & \text{if position } n \text{ is IBD} = 0 \text{ in two siblings} \end{cases}$$

IBD status π_n is not observable, but we do observe the IBS status X_n

$$X_n = \begin{cases} 2 & \text{if position } n \text{ is IBS} = 2 \text{ in two siblings} \\ 1 & \text{if position } n \text{ is IBS} = 1 \text{ in two siblings} \\ 0 & \text{if position } n \text{ is IBS} = 0 \text{ in two siblings} \end{cases}$$

where the describes the number of shared alleles between two affected siblings. In consanguineous families, the shared alleles will be homozygous, but in non-consanguineous families, the shared alleles may be heterozygous, but identically shared in each sibling.

6.5.1 First-order Transition Probabilities using IBD = 0, 1, 2

Consider two siblings. A locus n can be $\pi_n = 0$, $\pi_n = 1$ and $\pi_n = 2$ with probabilities

$$P[\pi_n = 0] = \frac{1}{4} \quad P[\pi_n = 1] = \frac{1}{2} \quad P[\pi_n = 2] = \frac{1}{4}$$

Using Ψ_m and Ψ_p which are functions of the sex-specific recombination rates (θ_m,θ_p):

$$\begin{aligned} \Psi_m &= \theta_m^2 + (1-\theta_m)^2 \\ \Psi_p &= \theta_p^2 + (1-\theta_p)^2 \end{aligned}$$

we can compute the transition probabilities $P[\pi_n = j|\pi_{n-1} = i]$ between loci n and $n-1$ using Ψ_m and Ψ_p. The transition probabilities are given in Table 6.2.
The joint probabilities are given by

$$P[\pi_n = j, \pi_{n-1} = i] = P[\pi_n = j|\pi_{n-1} = i]P[\pi_{n-1} = i]$$

Table 6.2 : Transition Probabilities $P[\pi_n = j|\pi_{n-1} = i]$ for IBD status (IBD = 0, 1, 2) [Feng et al., 2005]

	$\pi_n = 0$	$\pi_n = 1$	$\pi_n = 2$
$\pi_{n-1} = 0$	$\Psi_m\Psi_p$	$\Psi_m(1-\Psi_p) + \Psi_p(1-\Psi_m)$	$(1-\Psi_m)(1-\Psi_p)$
$\pi_{n-1} = 1$	$\frac{1}{2}(\Psi_m(1-\Psi_p) + \Psi_p(1-\Psi_m))$	$\Psi_m\Psi_p + (1-\Psi_m)(1-\Psi_p)$	$\frac{1}{2}(\Psi_m(1-\Psi_p) + \Psi_p(1-\Psi_m))$
$\pi_{n-1} = 2$	$(1-\Psi_m)(1-\Psi_p)$	$\Psi_m(1-\Psi_p) + \Psi_p(1-\Psi_m)$	$\Psi_m\Psi_p$

6.5.2 First-order Conditional Emissions using IBD = 0, 1, 2

First-order conditional emission probabilities are given by

$$e_{(X_{n-1},\pi_n)}(X_n) = P[X_n|X_{n-1},\pi_n]$$

To estimate these emission probabilities, we used the same procedure as in Section 6.3.2. The emission probabilities conditional on IBD $\neq 2$ in Models C and D also varied as a function of minor allele frequency at position n. There were 27 emission probabilities in this HMM and did provide the plots for brevity.

6.6 Application to simulated exome sequencing data

6.6.1 Simulation Study: root mean squared error

To assess the performance of the models in Table 6.1, we used the simulated families discussed in Section 6.1.1 because the IBD status is known. Using each family, we predicted the IBD regions using each model. We calculated root mean squared error (RMSE) using the true IBD status (π_n) and the predicted IBD status ($\hat{\pi}_n$) averaged over all N positions within a given family.

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(\hat{\pi}_n - \pi_n)^2}$$

After obtaining a family-specific RMSE estimate for each model, we averaged across the families to compare the models. First, we assessed the RMSE using emission probabilities which did not vary as a function of MAF (Figure 6.4). The figure shows Models C and D which use a first-order HMM with conditional emission probabilities using the IBD status $\in \{0, 1, 2\}$ has the smallest root MSE.

Next, we assessed the root MSE using emission probabilities as a function of MAF

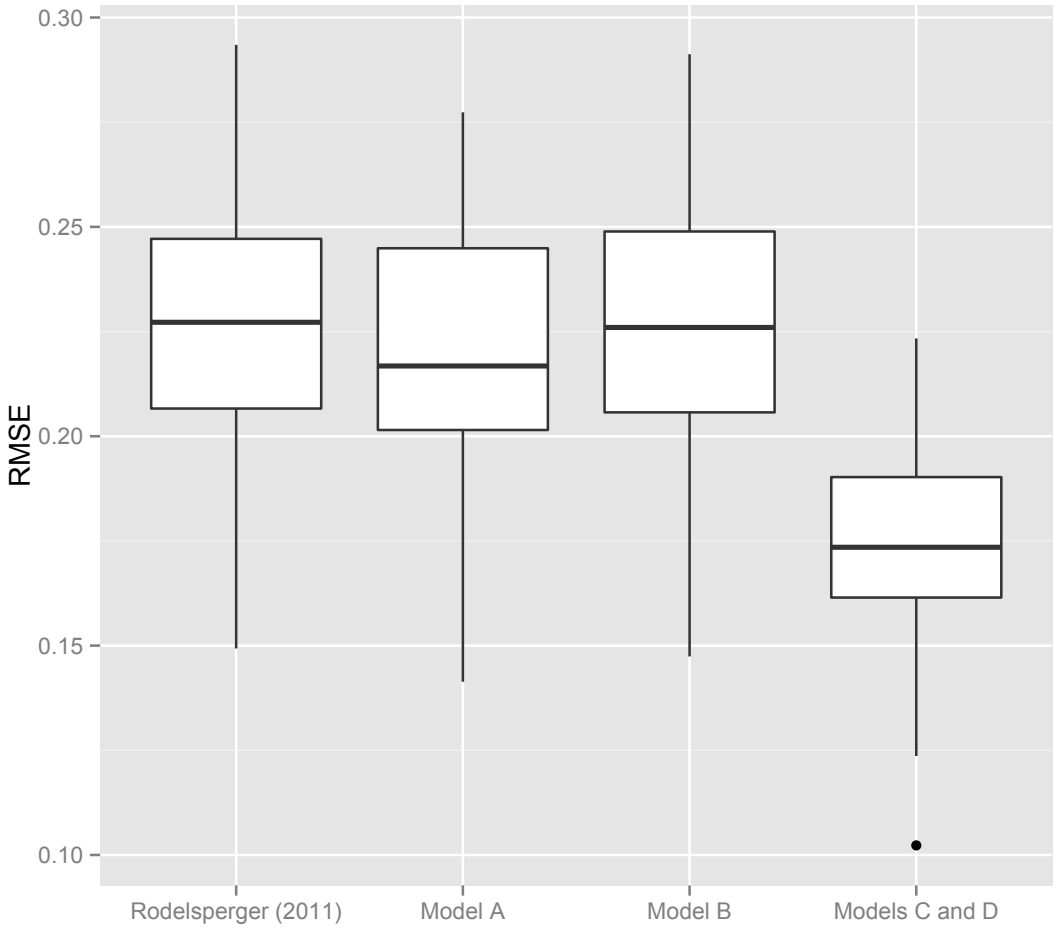

Figure 6.6 : A comparison of RMSE estimates (averaged over 100 simulated families) using the four models described in Table 6.1 without varyng MAF.

(Figure 6.5). We obtained MAF information from wANNOVAR for 100 families. In this case, figure shows Models C and D again has the smallest root MSE.

Table 6.3 contains the mean RMSE estimates for the models in Table 6.1 when considering MAF and not considering MAF in the emission probabilities.

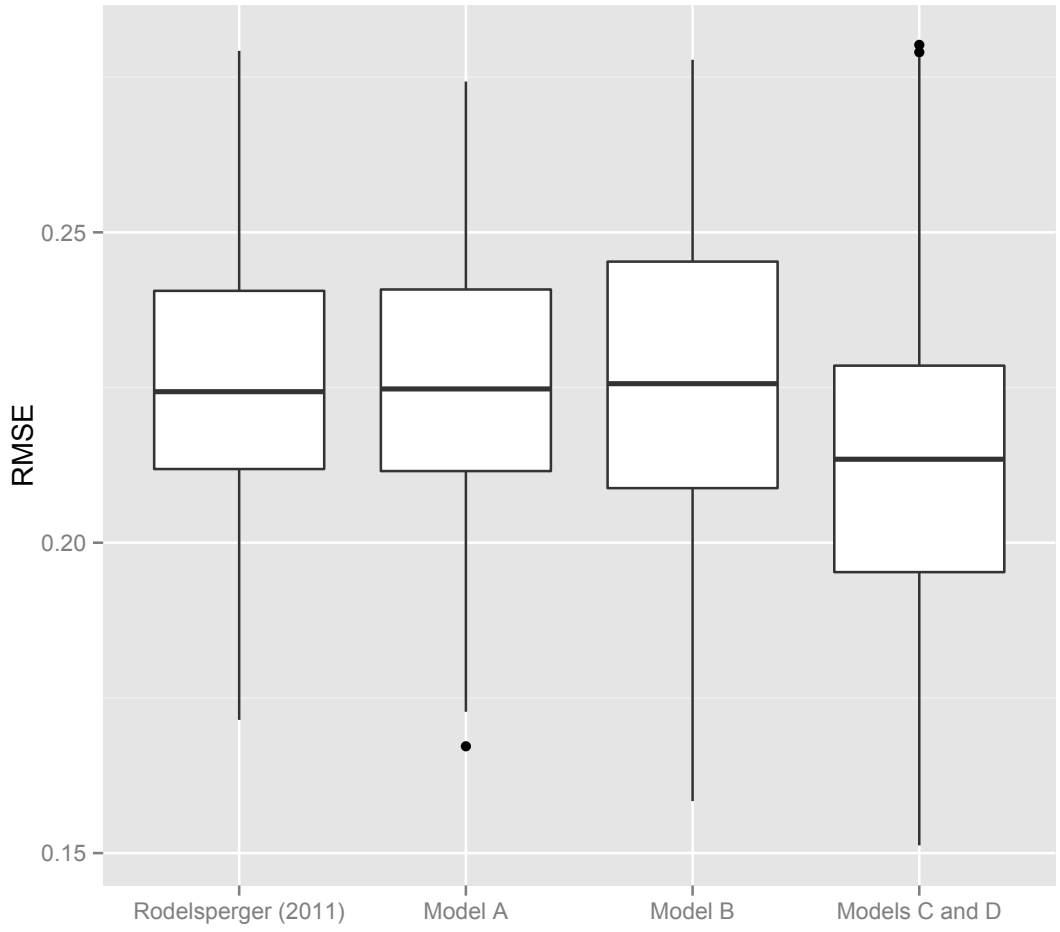

Figure 6.7 : A comparison of root MSE estimates (averaged over 100 simulated families) using the four models described in Table 6.1 varying MAF.

6.6.2 Visualizing regions of IBD

All inhomogeneous HMMs listed in Table 6.1 were tested using a set of exome sequencing data for a simulated family introduced in Section 6.1.1. The Viterbi algorithms were implemented and results for Chromosome 1 in a given simulated family are reported in Figure 6.8. The Viterbi predictions from Models C and D (in pink) most accurately predict the true IBD status (in black).

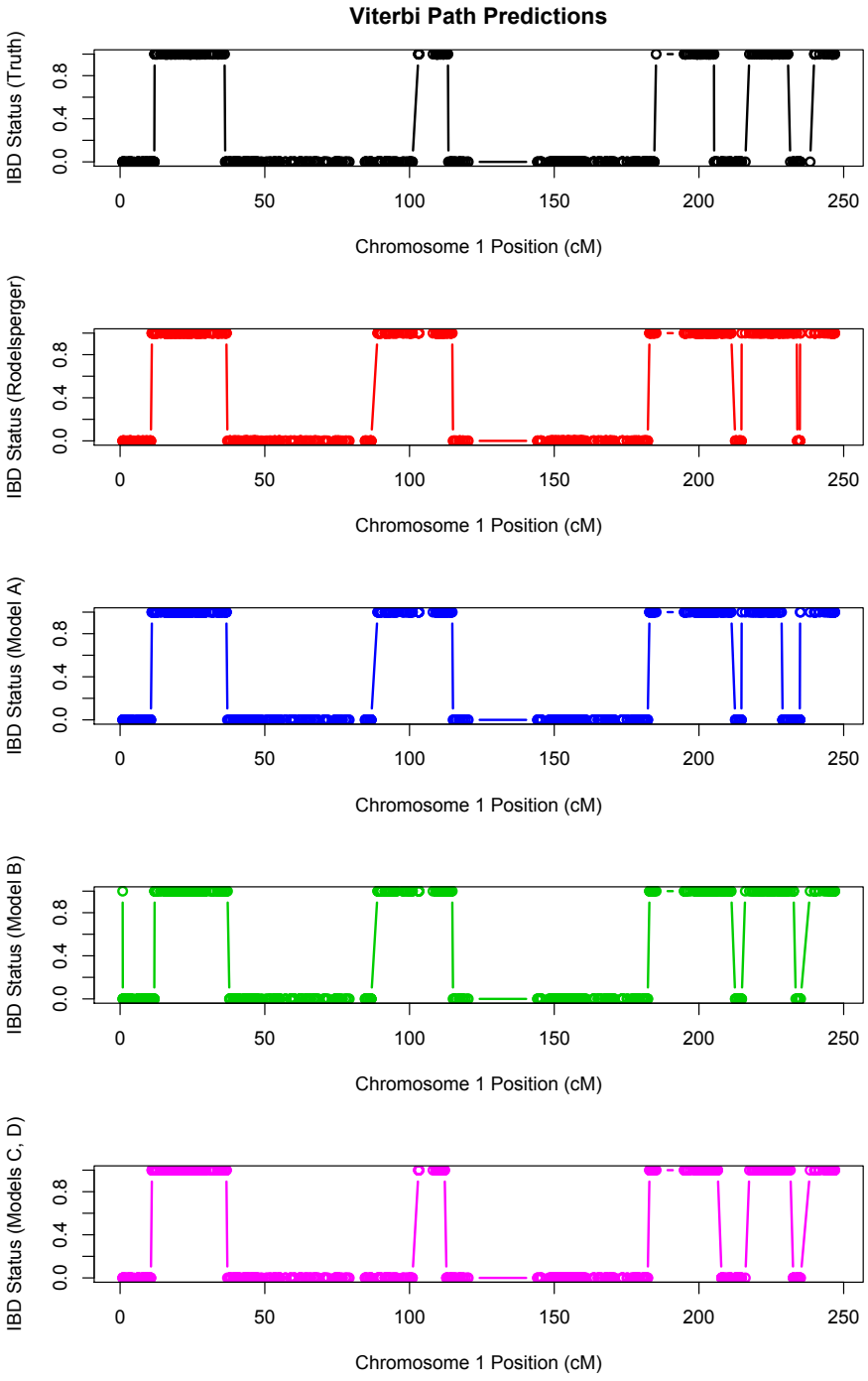

Figure 6.8 : Viterbi predictions for Chromosome 1 from a simulated family: True IBD status (black), Rodelsperger Model predictions (red), Model A predictions (blue), Model B predictions (green), Models predictions C and D (pink).

Table 6.3 : A comparison of mean RMSE estimates (averaged over 100 simulated families) using the four models described in Table 6.1 without considering MAF.

	Rodelsperger Model	Model A	Model B	Models C and D
No MAF varying	0.2256	0.2202	0.2281	0.1730
MAF varying	0.2250	0.2256	0.2262	0.2131

The marginal posterior probability of being an IBD=2 region is given in Figure 6.9. The marginal probability of being in an IBD = 2 regions from Models C and D (in pink) follows more closely the true IBD status (in black). Model B (second-order HMM) results a finer structure of the marginal probability of being in an IBD = 2 region than compared to the other HMMs. At the positions where the HMMs disagree, the genotypes of the two siblings were checked to see if a difference in the genotypes between the siblings were causing the finer structure in the second-order HMM. There did not seem to be any correlation between where the HMMs differed in their marginal posterior probabilities and where the genotypes differed between siblings.

6.7 Applications to human exome sequencing data

The inhomogeneous HMMs were tested on a pair of siblings from an acute lymphoblastic leukemia family. We compared the inhomogeneous first-order HMM developed [Rödelsperger et al., 2011] to Models A, B, C and D.

The Viterbi predictions for chromosome 1 (cM) predicting the IBD = 2 regions in the two siblings from the acute lymphoblastic family are shown in Figure 6.10. The marginal probability of being an IBD = 2 region is given in Figure 6.11. As there true IBD status is not known, we cannot compare the Viterbi predictions to the known

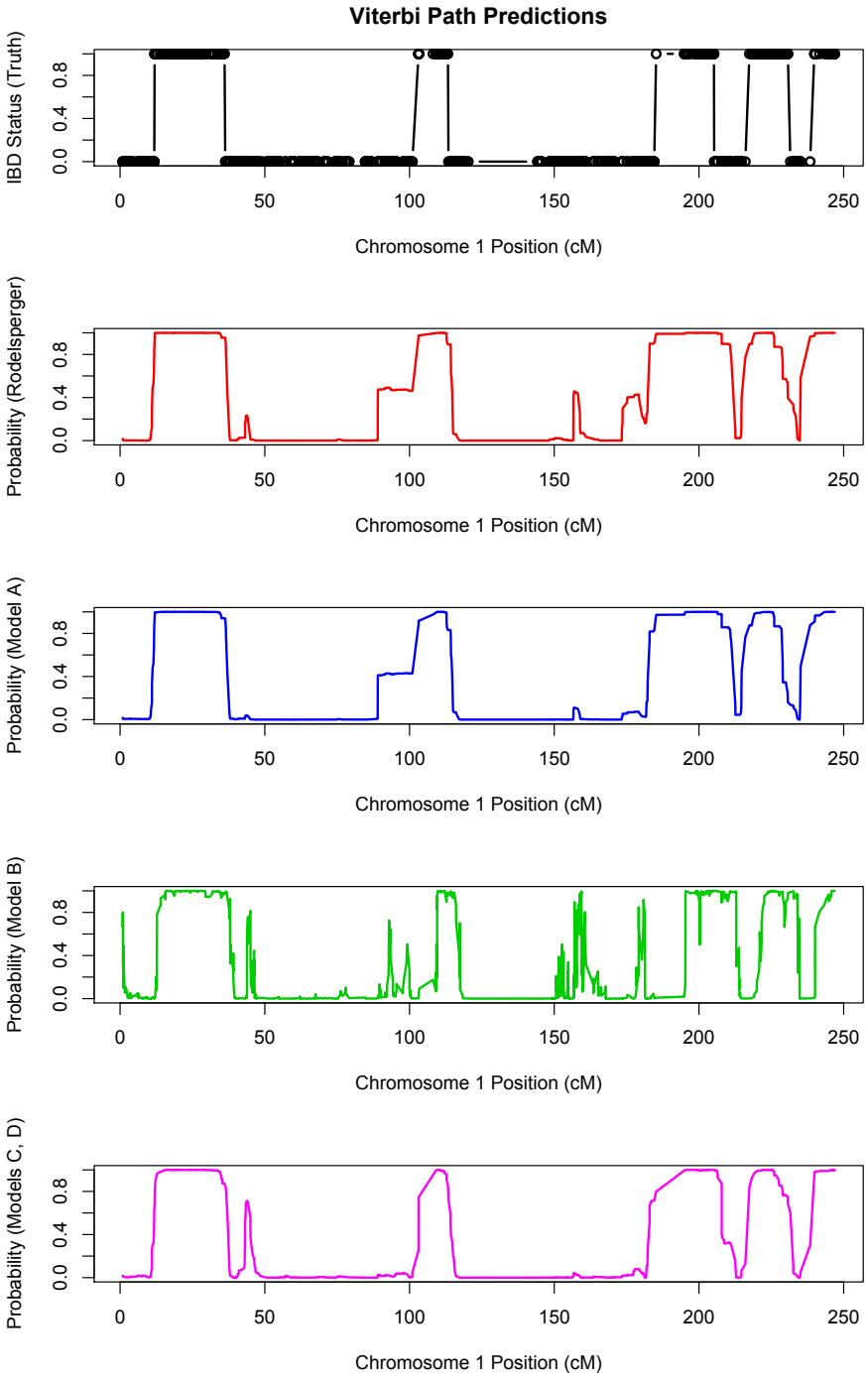

Figure 6.9 : Marginal posterior probability of being IBD = 2 in Chromosome 1 using a simulated family: True IBD status (black), Rodelsperger Model probability (red), Model A probability (blue), Model B probability (green), Models probability C and D (pink).

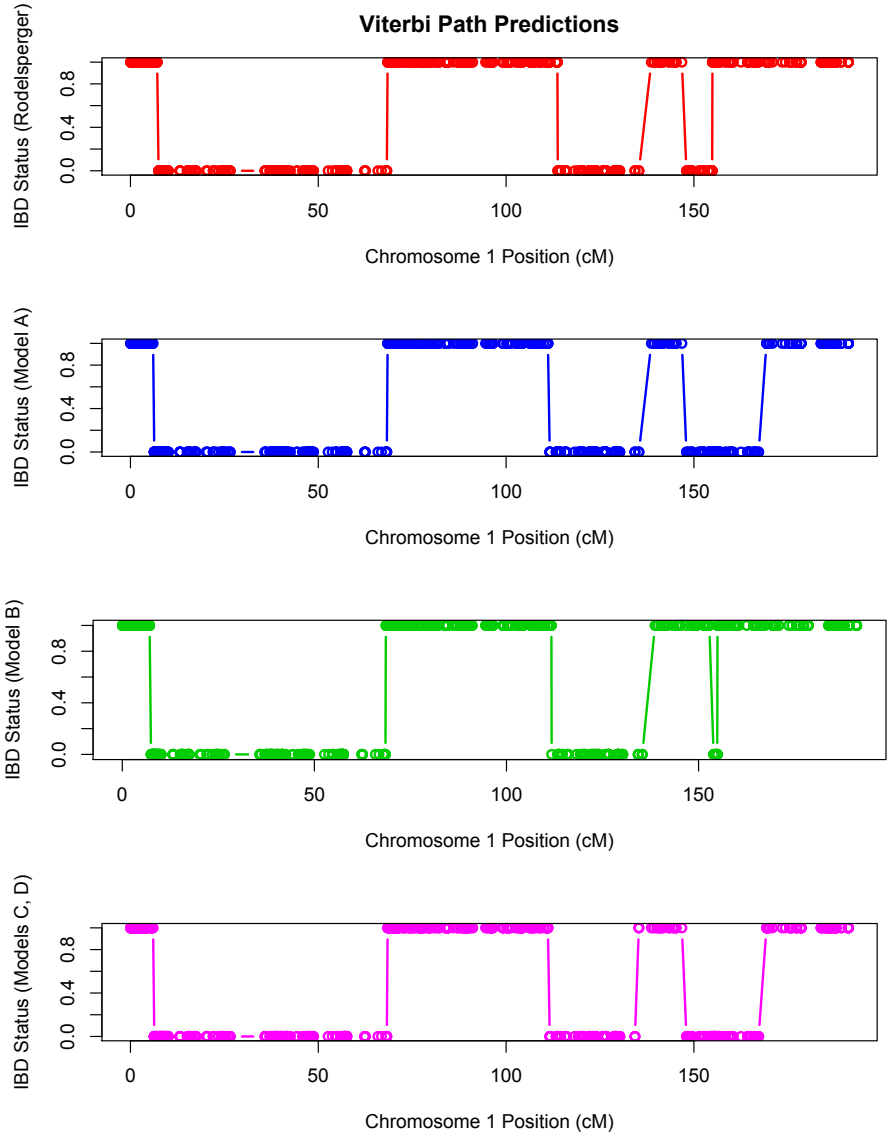

Figure 6.10 : Viterbi predictions for Chromosome 1 from an acute lymphoblastic leukemia family: Rodelsperger et al. (2011) Model predictions (red), Model A predictions (blue), Model B predictions (green), Models predictions C and D (pink).

IBD status.

6.8 Discussion

In this chapter we considered several inhomogeneous HMMs to predict regions of IBD using whole-exome sequencing data. We extended a previously developed first-order HMM to a first-order HMM with conditional emission probabilities and a second-order HMM to explore the second-order dependence structure between the observed variant calls in siblings. We also showed the emission probabilities used in the HMMs vary as a function of minor allele frequency. Interestingly, using MAF does not increase the accuracy of the HMMs.

We considered additional HMMs such a second-order HMM with conditional emissions defined using the IBD status IBD $= 2$, IBD $\neq 2$ and IBD $\in \{0, 1, 2\}$. For these models, we did derive an second-order iterative Viterbi-type algorithm (similar to the one developed in Section 6.3). The joint distribution of X and π is then given by

$$\begin{aligned} P(X, \pi) &= P[X|\pi]P[\pi] = E_L \Pi_L \\ &= E_1 \Pi_1 \prod_{n=2}^{L} P[X_n|X_{n-1}, \pi_n, \pi_{n-1}]P[\pi_n|\pi_{n-1}, \pi_{n-2}] \\ &= E_1 \Pi_1 \prod_{n=2}^{L} e_{(X_{n-1}, \pi_n, \pi_{n-1})}(X_n) P_{\pi_n, \pi_{n-1}, \pi_{n-2}} \end{aligned}$$

where $e_{(X_{n-1}, \pi_n, \pi_{n-1})}(X_n)$ is a conditional emission probability for the nth position and $P_{\pi_n, \pi_{n-1}, \pi_{n-2}}$ is the transition probability at the nth step given the two previous steps. The general steps for this second-order iterative Viterbi algorithm were as follows. Let

$$v_{\pi_0}(\pi_1, \pi_0) = E_1 \Pi_1 = P[X_1|\pi_1, \pi_0]P[\pi_1|\pi_0]P[\pi_0]$$

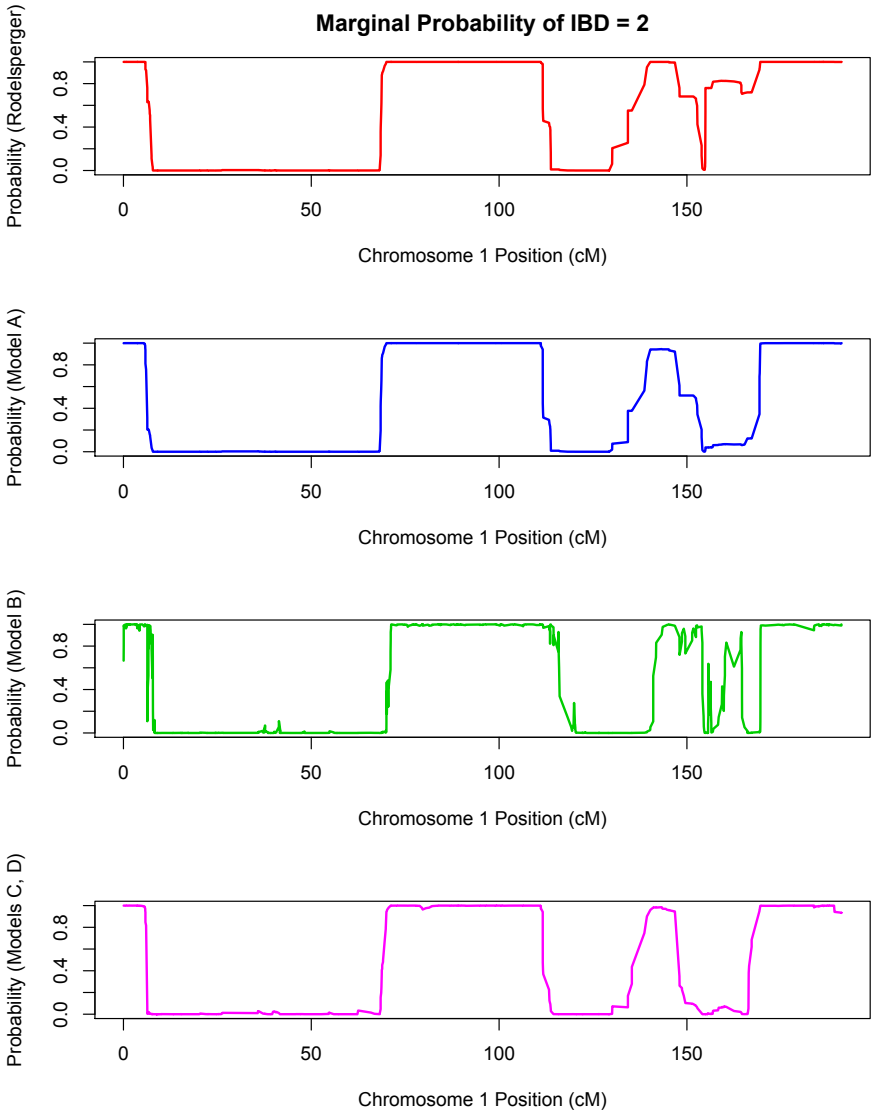

Figure 6.11 : Marginal posterior probability of being IBD = 2 in Chromosome 1 from an acute lymphoblastic leukemia family: Rodelsperger et al. (2011) Model probability (red), Model A probability (blue), Model B probability (green), Models probability C and D (pink).

then for $n = 1, \ldots L-1$,

$$v_{\pi_n}(\pi_{n+1}, \pi_n) = P[X_{n+1}|X_n, \pi_{n+1}, \pi_n] \max_{\pi_{n-1}}\{\ P[\pi_{n+1}|\pi_n, \pi_{n-1}] \times v_{\pi_{n-1}}(\pi_n, \pi_{n-1})\ \}$$

and for $n = L$,

$$v_{\pi_L}(\pi_L) = \max_{\pi_{L-1}}\{\ v_{\pi_{L-1}}(\pi_L, \pi_{L-1})\ \}$$

giving the final maximization

$$\max_{\pi} P(X, \pi) = \max_{\pi_L}\{\ v_{\pi_L}(\pi_L)\ \}$$

This model was not comparable in accuracy, so we excluded it from the results in this thesis.

We also considered computing the conditional emission probabilities using an previously developed approach [Albrechtsen et al., 2009, Han and Abney, 2011] which was developed for dense genotype data. The goal was to incorporate LD in IBD predictions. We faced several difficulties with this approach including the requirement of having haplotype frequencies to compute the conditional emission probabilities. Therefore, we decided to use a simulation-based approach and empirical frequencies to estimate the emission probabilities used in this thesis.

Chapter 7

Conclusions

In this thesis, I have developed probabilistic models in application for genetic and genomic data containing missing or unobservable information. In particular, I focused on mixture models and hidden Markov models which are used to make inference on unobserved information using some observed information. The main contributions of this thesis are the development of two probabilistic models: (1) a mixture model to estimate a unified posterior probability of functionality of missense mutations discussed in Chapter 4 and 5 and (2) a HMM to identify co-inherited regions in the exomes of related individual discussed in Chapter 6.

I briefly introduced mixture models and hidden Markov models in Chapter 2 and gave several examples of applications since the late 1960s and early 1970s. These models have frequently been estimated using the Expectation-Maximization (EM) algorithm. This method was introduced in Section 2.2 including how to estimate confidence intervals using the estimates from the EM algorithm. I briefly discussed the genetic architecture of diseases in Section 2.5. While Mendelian diseases are influenced by low-frequency high-risk variants, common diseases also have been argued to be caused by rare variants (CDRV hypothesis) with low to moderate effect sizes or by common variants (CDCV) with moderate effect sizes. Identifying these rare variants in Mendelian or common diseases is a difficult task. This thesis introduces new methods to identify the regions containing these disease mutations and to characterize the mutations found in the region before biological studies are performed (such as

functional assays).

Predicting the impact of missense mutations on protein function is an important factor in identifying and determining the clinical importance of disease susceptibility mutations. In Chapter 3, we performed a study investigating how functional predictions of missense mutations vary between using different *in silico* methods, and also vary using different protein sequence alignments. I showed that many computational or *in silico* methods have been developed to predict the functionality of missense mutations, but surprisingly there is a high degree of disagreement among the predictions produced by these methods even though the majority of these methods base their predictions on similar information (the use of evolutionary conservation as a measure of pathogenicity). This causes a great difficulty to researchers who use these algorithms as a way of filtering for mutations of interest from next-generation sequencing data because they will get different sets of final candidate variants based on employing different algorithms and sequence alignments. I showed that even when considering the same sequence alignment, the algorithms may make differing predictions. I note that Section 3.1 essentially contains the same information as the paper we published in Human Mutation [Hicks et al., 2011] which was recently highlighted and discussed in Nature [Baker, 2012]. From this study, I reviewed possible reasons for the disagreement between predictions of functionality and provide an example of the degree of disagreement in Section 3.2.

Another approach of identifying mutations impacting protein function may be to use hidden Markov models to gain position specific information about the functionality of missense mutations. Currently, these algorithms make predictions at each mutation of interest independently and do not consider the functionality of surrounding mutations. If an algorithm predicts a mutation as deleterious at a given position,

then it may be in a functionally relevant portion region of the gene that is evolutionary conserved. Given the first mutation was predicted to be deleterious, it is expected that a second mutation very close to the first would also be predicted deleterious while a second mutation farther way would be less correlated with the functionality of the first mutation.

As future work, I would like to investigate this novel idea of using inhomogenous first-order HMMs to make inference on the true functionality of missense mutations using the observed predictions from the bioinformatic algorithms (e.g. PolyPhen-2, SIFT). The idea is to model the pathogeneticity of mutations along a given gene to increase sensitivity and specificity of the computational methods because when not enough information is known about a particular mutation, the HMM may make inference using the information known about a mutation within a close distance on a given gene if available. The goal would be to predict the unobserved true functionality of the mutation as either truly neutral or truly deleterious given the observed functional predictions. I believe this approach could increase the sensitivity and specificity of the predictions by using inhomogeneous transition probabilities at each mutation that account for the distance between the mutations in a given gene. Using the notation defined in Section 1.1.3, the true functionality status would be provided by the unobservable Markov chain π_n and the observed predictions from a given computational method would be given by X_n.

One of the barriers to this approach is the lack of 'gold standard' for this type of data. There is lack of exome-scale set of missense mutations with known functional impact on protein function. One reason is that the functionality of exomic data is not known a priori so we would not able to directly compare sensitivity and specificity values for the PolyPhen-2 predictions and Viterbi predictions. Another great

challenge is how to create a standardized approach that can use any of the existing bioinformatic algorithms semi-automatically on either the well-characterized sets of mutations or mutations called from exome sequencing.

Because the functional predictions from available *in silico* methods often have a high degree of disagreement, we developed a mixture model which estimates a unified posterior probability of functionality or pathogenicity. In Chapter 4, two statistical models based on the capture-recapture paradigm were developed which combine the discordant functional predictions in a statistically rigorous manner. Our models estimate a unified posterior probability of functionality or pathogenicity for each missense mutation. Unlike previous methods, our probabilistic approach requires no training set or calibration and estimates the accuracy (sensitivity and specificity) of each individual *in silico* method in the absence of a gold standard by taking advantage of the fact these methods disagree. Compared to previous attempts to combine predictions of functionality which weight the predictions by normalized scores or allele frequencies, our approach weights the functional predictions by the estimated accuracy of each method. In Section 4.2 the two models referred to as postMUT and postMUT (simple) were introduced and the parameter estimates for the Expectation-Maximization algorithm were derived. In Section 4.3 we showed our estimates of sensitivity and specificity of the *in silico* algorithms (without employing a gold standard) match the estimates of sensitivity and specificity when a gold standard is available. The posterior probability of pathogenicity introduced in this chapter is a statistical tool scalable to the exome which may be used to infer the functionality of missense mutations and can be easily incorporated in downstream analyses such as disease gene prioritization tools ultimately inferring candidate genes.

In Chapter 5, we performed a set of simulations assessing the bias and mean

squared error of the parameter estimates from our postMUT (simple) and postMUT models. We considered various scenarios such as varying the sensitivity and specificity of each *in silico* algorithm and varying the number of algorithms considered. This is important because the posterior probabilities weight the functional predictions based on the accuracy of each algorithm. We showed how the confidence regions of the sensitivity and specificity estimates vary depending on the overall proportion of deleterious mutations in set considered.

In Chapter 6, we developed several hidden Markov models (HMMs) to identify regions of IBD using the observed IBS status from whole-exome sequencing data. We considered several extensions of a previously developed first-order inhomogeneous HMM that identifies regions of IBD in siblings that can be used a filter in search of finding the casual variant for a given disease. A comparison of the models considered is given in Table 6.1. We explored the idea of using conditional emission probabilities (depends on both IBD and IBS status) varying as a function of minor allele frequency and a second-order HMM which models the second-order dependence structure between observed variant calls. A non-trivial challenge is how to develop accurate models for human Mendelian disorders concerning recombination patterns while accounting for any dependence structure between observed variant calls in related individuals.

One possible way to improve these HMMs is to estimate the emission probabilities using genotype and haplotype frequencies as opposed to our approach of using simulation. We did investigate this approach which was previously discussed [Han and Abney, 2011], but decided on a simulation-based approach because the combinatorics quickly became very complicated. For example, one HMM we considered had 81 different possible combinations of IBD and IBS states making it almost in-

tractable to approach from a probabilistic view.

We applied these models to whole-exome sequencing data and showed how these models can be used to identify disease susceptibility mutations. We assessed the accuracy of the models by simulating whole-exome sequencing data discussed in Section 6.1.1. Using the known IBD status from the simulated families, we were able to compare the root mean squared error from each each model averaged over a set of families. We showed a first-order HMM with conditional emission probabilities defined using the hidden IBD status $\in \{0, 1, 2\}$ has smaller root mean squared error compared to the first-order HMM previously developed [Rödelsperger et al., 2011]. The second-order HMM performed comparably to the first-order HMM suggesting a second-order dependence structure does not increase the accuracy of predicting IBD regions. We also applied these HMMs to a set a unpublished real human exome sequencing data in Section 6.7 which was discussed in Section 6.1.2. We predicted regions of IBD on a pair of siblings in this family with an autosomal dominant disorder.

As disease-gene identification projects increasingly use next-generation sequencing, the probabilistic models developed in this thesis help identify and associate relevant disease-causing mutations with human disorders. The purpose of this thesis is to demonstrate that probabilistic models can contribute to more accurate and dependable inference based on genetic and genomic data with missing information.

Bibliography

[Abkevich et al., 2003] Abkevich, V., Zharkikh, A., Deffenbaugh, A., Frank, D., Chen, Y., Shattuck, D., Skolnick, M., Gutin, A., and Tavtigian, S. (2003). Analysis of missense variation in human BRCA1 in the context of interspecific sequence variation. *Journal of Medical Genetics*, 41:492–507.

[Acharya and Nagarajaram, 2012] Acharya, V. and Nagarajaram, H. A. (2012). Hansa: An automated method for discriminating disease and neutral human nsSNPs. *Hum Mutat*, 33(2):332–7.

[Adzhubei et al., 2010] Adzhubei, I. A., Schmidt, S., Peshkin, L., Ramensky, V. E., Gerasimova, A., Bork, P., Kondrashov, A. S., and Sunyaev, S. R. (2010). A method and server for predicting damaging missense mutations. *Nat Methods*, 7(4):248–9.

[Agresti, 2002] Agresti, A. (2002). *Categorical Data Analysis*. John Wiley and Sons, Hoboken, NJ, 2nd edition.

[Albert and Chib, 1993] Albert, J. H. and Chib, S. (1993). Bayes inference via Gibbs sampling of autoregressive time series subject to Markov mean and variance shifts. *Journal of Business and Economic Statistics*, 11(1):1–15.

[Albrechtsen et al., 2009] Albrechtsen, A., Sand Korneliussen, T., Moltke, I., van Overseem Hansen, T., Nielsen, F. C., and Nielsen, R. (2009). Relatedness mapping and tracts of relatedness for genome-wide data in the presence of linkage disequilibrium. *Genet Epi*, 33(3):266–74.

[Asai et al., 1993] Asai, K., Hayamizu, S., and Handa, K. (1993). Prediction of protein secondary structure by the hidden Markov model. *Comput Appl Biosci*, 9(2):141–6.

[Bae et al., 2008] Bae, K., Mallick, B. K., and Elsik, C. G. (2008). Prediction of protein interdomain linker regions by a nonstationary hidden Markov model. *J Am Stat Assoc*, 103(483):1085–1098.

[Baker, 2012] Baker, M. (2012). Functional genomics: The changes that count. *Nature*, 482(7384):257, 259–62.

[Balasubramanian et al., 2005] Balasubramanian, S., Xia, Y., Freinkman, E., and Gerstein, M. (2005). Sequence variation in G-protein-coupled receptors: analysis of single nucleotide polymorphisms. *Nucleic Acids Res*, 33(5):1710–1721.

[Baldi et al., 1994] Baldi, P., Chauvin, Y., Hunkapiller, T., and McClure, M. A. (1994). Hidden Markov models of biological primary sequence information. *Proc Natl Acad Sci U S A*, 91(3):1059–63.

[Balding et al., 2007] Balding, D. J., Bishop, M., and Cannings, C., editors (2007). *Handbook of Statistical Genetics*. Wiley-Interscience, 3rd edition.

[Bao and Cui, 2005] Bao, L. and Cui, Y. (2005). Prediction of the phenotypic effects of non-synonymous single nucleotide polymorphisms using structural and evolutionary information. *Bioinformatics (Oxford, England)*, 21(10):2185–2190.

[Benaglia et al., 2009] Benaglia, T., Chauveau, D., Hunder, D. R., and Young, D. (2009). mixtools: An R packages for analyzing finite mixture models. *Journal of Statistical Software*, 32(6):1–29.

[Bird, 1987] Bird, A. P. (1987). CpG islands as gene markers in the vertebrate nucleus. *Trends in Genetics*, 3:342–347.

[Blimes, 1998] Blimes, J. (1998). A gentle tutorial of the EM algorithm and its application to parameter estimation for gaussian mixture and hidden Markov models. Technical report, UC Berkeley.

[Bodmer and Bonilla, 2008] Bodmer, W. and Bonilla, C. (2008). Common and rare variants in multifactorial susceptibility to common diseases. *Nat Genet*, 40(6):695–701.

[Brennan et al., 2007] Brennan, D. J., Kelly, C., Rexhepaj, E., Dervan, P. A., Duffy, M. J., and Gallagher, W. M. (2007). Contribution of DNA and tissue microarray technology to the identification and validation of biomarkers and personalised medicine in breast cancer. *Cancer Genomics Proteomics*, 4(3):121–34.

[Brigo and Mercurio, 2002] Brigo, D. and Mercurio, F. (2002). Lognormal-mixture dynamics and calibration to market volatility smiles. *International Journal of Theoretical and Applied Finance*, 5(4):427.

[Bucher et al., 1996] Bucher, P., Karplus, K., Moeri, N., and Hofmann, K. (1996). A flexible motif search technique based on generalized profiles. *Comput Chem*, 20(1):3–23.

[Burge and Karlin, 1997] Burge, C. and Karlin, S. (1997). Prediction of complete gene structures in human genomic DNA. *J Mol Biol*, 268(1):78–94.

[Carter and Falconer, 1951] Carter, T. C. and Falconer, D. S. (1951). Stocks for detecing linkage in the mouse, and the theory of their design. *Journal of Genetics*, 50(2):307–323.

[Chan et al., 2007] Chan, P. A., Duraisamy, S., Miller, P. J., Newell, J. A., McBride, C., Bond, J. P., Raevaara, T., Ollila, S., Nystrm, M., Grimm, A. J., Christodoulou, J., Oetting, W. S., and Greenblatt, M. S. (2007). Interpreting missense variants: comparing computational methods in human disease genes CDKN2A, MLH1, MSH2, MECP2, and tyrosinase (TYR). *Human Mutation*, 28(7):683–693.

[Chang and Wang, 2012] Chang, X. and Wang, K. (2012). wANNOVAR: annotating genetic variants for personal genomes via the web. *J Med Genet*, 49(7):433–6.

[Chao et al., 2008] Chao, E. C., Velasquez, J. L., Witherspoon, M. S. L., Rozek, L. S., Peel, D., Ng, P., Gruber, S. B., Watson, P., Rennert, G., Anton-Culver, H., Lynch, H., and Lipkin, S. M. (2008). Accurate classification of MLH1/MSH2 missense variants with multivariate analysis of protein polymorphisms-mismatch repair (MAPP-MMR). *Human Mutation*, 29(6):852–860.

[Choi et al., 2009] Choi, M., Scholl, U. I., Ji, W., Liu, T., Tikhonova, I. R., Zumbo, P., Nayir, A., Bakkaloğlu, A., Ozen, S., Sanjad, S., Nelson-Williams, C., Farhi, A., Mane, S., and Lifton, R. P. (2009). Genetic diagnosis by whole exome capture and massively parallel DNA sequencing. *Proc Natl Acad Sci USA*, 106(45):19096–101.

[Chun and Fay, 2009] Chun, S. and Fay, J. C. (2009). Identification of deleterious mutations within three human genomes. *Genome Res*, 19(9):1553–61.

[Churchill, 1989] Churchill, G. A. (1989). Stochastic models for heterogeneous DNA sequences. *Bull Math Biol*, 51(1):79–94.

[Cirulli and Goldstein, 2010] Cirulli, E. T. and Goldstein, D. B. (2010). Uncovering the roles of rare variants in common disease through whole-genome sequencing. *Nat Rev Genet*, 11(6):415–25.

[Conlon, 2008] Conlon, E. M. (2008). A bayesian mixture model for metaanalysis of microarray studies. *Funct Integr Genomics*, 8(1):43–53.

[Cooper and Shendure, 2011] Cooper, G. M. and Shendure, J. (2011). Needles in stacks of needles: finding disease-causal variants in a wealth of genomic data. *Nat Rev Genet*, 12(9):628–40.

[Dempster et al., 1977] Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society Series B (Methodological)*, 39(1):1–38.

[Djuric and Chun, 2002] Djuric, P. M. and Chun, J.-H. (2002). An MCMC sampling approch to estimation of nonstationary hidden Markov models. *IEEE Transactions on Signal Processing*, 50(5):1113–1123.

[Drmanac et al., 2010] Drmanac, R., Sparks, A. B., Callow, M. J., Halpern, A. L., Burns, N. L., Kermani, B. G., Carnevali, P., Nazarenko, I., Nilsen, G. B., Yeung, G., Dahl, F., Fernandez, A., Staker, B., Pant, K. P., Baccash, J., Borcherding, A. P., Brownley, A., Cedeno, R., Chen, L., Chernikoff, D., Cheung, A., Chirita, R., Curson, B., Ebert, J. C., Hacker, C. R., Hartlage, R., Hauser, B., Huang, S., Jiang, Y., Karpinchyk, V., Koenig, M., Kong, C., Landers, T., Le, C., Liu, J., McBride, C. E., Morenzoni, M., Morey, R. E., Mutch, K., Perazich, H., Perry, K., Peters, B. A., Peterson, J., Pethiyagoda, C. L., Pothuraju, K., Richter, C., Rosenbaum, A. M., Roy, S., Shafto, J., Sharanhovich, U., Shannon, K. W., Sheppy, C. G., Sun, M., Thakuria, J. V., Tran, A., Vu, D., Zaranek, A. W., Wu, X., Drmanac, S., Oliphant, A. R., Banyai, W. C., Martin, B., Ballinger, D. G., Church, G. M., and Reid, C. A. (2010). Human genome sequencing using unchained base reads on self-assembling DNA nanoarrays. *Science*, 327(5961):78–81.

[Durbin et al., 1998] Durbin, R. M., Eddy, S. R., Krogh, A., and Mitchison, G. (1998). *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge University Press.

[Eddy, 1998] Eddy, S. R. (1998). Profile hidden Markov models. *Bioinformatics*, 14(9):755–63.

[Efron et al., 2001] Efron, B., Tibshirani, R., Storey, J. D., and Tusher, V. (2001). Empirical Bayes analysis of microarray experiment. *Journal of the American Statistical Association*, 96(456):1151–1160.

[Fawcett, 2006] Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recogn Lett*, 27:861–874.

[Felsenstein, 1979] Felsenstein, J. (1979). A mathematically tractable family of genetic mapping functions with different amounts of interference. *Genetics*, 91(4):769–75.

[Felsenstein, 1981] Felsenstein, J. (1981). Evolutionary trees from DNA sequences: a maximum likelihood approach. *J Mol Evol*, 17(6):368–76.

[Felsenstein and Churchill, 1996] Felsenstein, J. and Churchill, G. A. (1996). A hidden Markov model approach to variation among sites in rate of evolution. *Mol Biol Evol*, 13(1):93–104.

[Feng et al., 2005] Feng, Z. Z., Chen, J., and Thompson, M. E. (2005). The universal validity of the possible triangle constraint for affected sib pairs. *The Candian Journal of Statistics*, 33(2):297–310.

[Finn et al., 2010] Finn, R. D., Mistry, J., Tate, J., Coggill, P., Heger, A., Pollington, J. E., Gavin, O. L., Gunasekaran, P., Ceric, G., Forslund, K., Holm, L., Sonnhammer, E. L. L., Eddy, S. R., and Bateman, A. (2010). The pfam protein families database. *Nucleic Acids Res*, 38(Database issue):D211–22.

[Fraley and Raftery, 2012] Fraley, C. and Raftery, A. (June 2012). MCLUST version 4 for R: Normal mixture modeling for model-based clustering classification, and density estimation. Technical Report no. 597, Department of Statistics, University of Washington.

[Goldgar et al., 2004] Goldgar, D. E., Easton, D. F., Deffenbaugh, A. M., Monteiro, A. N. A., Tavtigian, S. V., and Couch, F. J. a. (2004). Integrated evaluation of DNA sequence variants of unknown clinical significance: application to BRCA1 and BRCA2. *American journal of human genetics*, 75(4):535–544.

[Goldman et al., 1996] Goldman, N., Thorne, J. L., and Jones, D. T. (1996). Using evolutionary trees in protein secondary structure prediction and other comparative sequence analyses. *J Mol Biol*, 263(2):196–208.

[Gonzaga-Jauregui et al., 2012] Gonzaga-Jauregui, C., Lupski, J. R., and Gibbs, R. A. (2012). Human genome sequencing in health and disease. *Annu Rev Med*, 63:35–61.

[González-Pérez and López-Bigas, 2011] González-Pérez, A. and López-Bigas, N. (2011). Improving the assessment of the outcome of nonsynonymous SNVs with a consensus deleteriousness score, Condel. *Am J Hum Genet*, 88(4):440–9.

[Gray et al., 2012] Gray, V. E., Kukurba, K. R., and Kumar, S. (2012). Performance of computational tools in evaluating the functional impact of laboratory-induced

amino acid mutations. *Bioinformatics*, 28(16):2093–6.

[Greenblatt et al., 2003] Greenblatt, M., Beaudet, J., Gump, J., Godin, K., Trombley, L., Koh, J., and Bond, J. (2003). Detailed computational study of p53 and p16: using evolutionary sequence analysis and disease-associated mutations to predict the functional consequences of allelic variants. *Oncogene*, 22(8):1150–1163.

[Greenblatt et al., 2008] Greenblatt, M. S., Brody, L. C., Foulkes, W. D., Genuardi, M., Hofstra, R. M. W., Olivier, M., Plon, S. E., Sijmons, R. H., Sinilnikova, O., Spurdle, A. B., and the IARC Unclassified Genetic Variants Working Group, f. (2008). Locus-specific databases and recommendations to strengthen their contribution to the classification of variants in cancer susceptibility genes. *Human Mutation*, 29(11):1273–1281.

[Guttorp, 1995] Guttorp, P. (1995). *Stochastic Modeling of Scientific Data*. Chapman and Hall.

[Haldane, 1919] Haldane, J. S. (1919). The combination of linkage values, and the calculation of distances between the loci of linked factors. *Journal of Genetics*, 8:299–309.

[Hamilton, 1989] Hamilton, J. D. (1989). A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica*, 57:357–384.

[Han and Abney, 2011] Han, L. and Abney, M. (2011). Identity by descent estimation with dense genome-wide genotype data. *Genet Epidemiol*, 35(6):557–67.

[Hanczar et al., 2010] Hanczar, B., Hua, J., Sima, C., Weinstein, J., Bittner, M., and Dougherty, E. R. (2010). Small-sample precision of ROC-related estimates. *Bioinformatics*, 26(6):822–30.

[Hastie et al., 2009] Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics, New York.

[Henderson et al., 1997] Henderson, J., Salzberg, S., and Fasman, K. H. (1997). Finding genes in DNA with a hidden Markov model. *J Comput Biol*, 4(2):127–41.

[Hicks et al., 2013] Hicks, S., Plon, S. E., and Kimmel, M. (2013). Statistical analysis of missense mutation classifiers. *Hum Mutat*, 34(2):405–6.

[Hicks et al., 2011] Hicks, S., Wheeler, D. A., Plon, S. E., and Kimmel, M. (2011). Prediction of missense mutation functionality depends on both the algorithm and sequence alignment employed. *Hum Mutat*, 32(6):661–8.

[Hogg and Tanis, 2006] Hogg, R. V. and Tanis, E. A. (2006). *Probability and Statistical Inference*. Pearson Prentice Hall, UPper Saddle River, NJ, 7th edition.

[Hughey and Krogh, 1996] Hughey, R. and Krogh, A. (1996). Hidden Markov models for sequence analysis: extension and analysis of the basic method. *Comput Appl Biosci*, 12(2):95–107.

[International HapMap Consortium, 2003] International HapMap Consortium (2003). The international HapMap project. *Nature*, 426(6968):789–96.

[Jaffe et al., 2011] Jaffe, A., Wojcik, G., Chu, A., Golozar, A., Maroo, A., Duggal, P., and Klein, A. P. (2011). Identification of functional genetic variation in exome sequence analysis. *BMC Proc*, 5 Suppl 9:S13.

[Jordan et al., 2010] Jordan, D. M., Ramensky, V. E., and Sunyaev, S. R. (2010). Human allelic variation: perspective from protein function, structure, and evolution.

Curr Opin Struct Biol, 20(3):342–50.

[Karchin, 2009] Karchin, R. (2009). Next generation tools for the annotation of human SNPs. *Briefings in Bioinformatics*, 10(1):35–52.

[Karchin et al., 2008] Karchin, R., Agarwal, M., Sali, A., Couch, F., and Beattie, M. S. (2008). Classifying variants of undetermined significance in BRCA2 with protein likelihood ratios. *Cancer Inform*, 6:203–16.

[Karlin and Liberman, 1978] Karlin, S. and Liberman, U. (1978). Classifications and comparisons of multilocus recombination distributions. *Proc Natl Acad Sci U S A*, 75(12):6332–6.

[Karplus et al., 1998] Karplus, K., Barrett, C., and Hughey, R. (1998). Hidden Markov models for detecting remote protein homologies. *Bioinformatics*, 14(10):846–56.

[Knapp and Chen, 2007] Knapp, K. and Chen, Y.-P. P. (2007). An evaluation of contemporary hidden Markov model genefinders with a predicted exon taxonomy. *Nucleic Acids Res*, 35(1):317–24.

[Kosambi, 1944] Kosambi, D. D. (1944). The estimation of map distance from recombination values. *Annals of Eugenics*, 12(3):172–175.

[Krogh, 1997] Krogh, A. (1997). Two methods for improving performance of an HMM and their application for gene finding. *Proc Int Conf Intell Syst Mol Biol*, 5:179–86.

[Krogh et al., 1994a] Krogh, A., Brown, M., Mian, I. S., Sjölander, K., and Haussler, D. (1994a). Hidden Markov models in computational biology. applications to protein modeling. *J Mol Biol*, 235(5):1501–31.

[Krogh et al., 1994b] Krogh, A., Mian, I. S., and Haussler, D. (1994b). A hidden Markov model that finds genes in E. coli DNA. *Nucleic Acids Res*, 22(22):4768–78.

[Kruglyak et al., 1996] Kruglyak, L., Daly, M. J., Reeve-Daly, M. P., and Lander, E. S. (1996). Parametric and nonparametric linkage analysis: a unified multipoint approach. *Am J Hum Genet*, 58(6):1347–63.

[Kryukov et al., 2007] Kryukov, G. V., Pennacchio, L. A., and Sunyaev, S. R. (2007). Most rare missense alleles are deleterious in humans: implications for complex disease and association studies. *Am J Hum Genet*, 80(4):727–39.

[Kulp et al., 1996] Kulp, D., Haussler, D., Reese, M. G., and Eeckman, F. H. (1996). A generalized hidden Markov model for the recognition of human genes in DNA. *Proc Int Conf Intell Syst Mol Biol*, 4:134–42.

[Lander and Green, 1987] Lander, E. S. and Green, P. (1987). Construction of multilocus genetic linkage maps in humans. *Proc Natl Acad Sci U S A*, 84(8):2363–7.

[Lange, 1995] Lange, K. (1995). A quasi-newton acceleration of the EM algorithm. *Statistica Sinica*, 5:1–18.

[Lawrence et al., 2009] Lawrence, M., Carey, V., and Gentleman, R. (2009). rtracklayer: an R package for interfacing with genome browsers. *Bioinformatics*, 25(14):1841–1842.

[Lee et al., 2000] Lee, M. L., Kuo, F. C., Whitmore, G. A., and Sklar, J. (2000). Importance of replication in microarray gene expression studies: statistical methods and evidence from repetitive cDNA hybridizations. *Proc Natl Acad Sci U S A*, 97(18):9834–9.

[Lehmann and Casella, 1998] Lehmann, E. and Casella, G. (1998). *Theory of Point Estimation*. Springer New York, second edition.

[Leutenegger et al., 2003] Leutenegger, A.-L., Prum, B., Génin, E., Verny, C., Lemainque, A., Clerget-Darpoux, F., and Thompson, E. A. (2003). Estimation of the inbreeding coefficient through use of genomic data. *Am J Hum Genet*, 73(3):516–23.

[Li et al., 2012] Li, M.-X., Gui, H.-S., Kwan, J. S. H., Bao, S.-Y., and Sham, P. C. (2012). A comprehensive framework for prioritizing variants in exome sequencing studies of mendelian diseases. *Nucleic Acids Res*, 40(7):e53.

[Liu et al., 2011] Liu, X., Jian, X., and Boerwinkle, E. (2011). dbNSFP: A lightweight database of human nonsynonymous SNPs and their functional predictions. *Hum Mutat*, 32(8):894–9.

[Lopes et al., 2012] Lopes, M. C., Joyce, C., Ritchie, G. R. S., John, S. L., Cunningham, F., Asimit, J., and Zeggini, E. (2012). A combined functional annotation score for non-synonymous variants. *Hum Hered*, 73(1):47–51.

[Louis, 1982] Louis, T. A. (1982). Finding the observed information matrix when using the EM algorithm. *Journal of the Royal Statistical Society Series B (Methodological)*, 44(2):226–233.

[Lukashin and Borodovsky, 1998] Lukashin, A. V. and Borodovsky, M. (1998). Genemark.HMM: new solutions for gene finding. *Nucleic Acids Res*, 26(4):1107–15.

[Lyon and Wang, 2012] Lyon, G. J. and Wang, K. (2012). Identifying disease mutations in genomic medicine settings: current challenges and how to accelerate progress. *Genome Med*, 4(7):58.

[Manolio et al., 2009] Manolio, T. A., Collins, F. S., Cox, N. J., Goldstein, D. B., Hindorff, L. A., Hunter, D. J., McCarthy, M. I., Ramos, E. M., Cardon, L. R., Chakravarti, A., Cho, J. H., Guttmacher, A. E., Kong, A., Kruglyak, L., Mardis, E., Rotimi, C. N., Slatkin, M., Valle, D., Whittemore, A. S., Boehnke, M., Clark, A. G., Eichler, E. E., Gibson, G., Haines, J. L., Mackay, T. F. C., McCarroll, S. A., and Visscher, P. M. (2009). Finding the missing heritability of complex diseases. *Nature*, 461(7265):747–53.

[Mathé et al., 2002] Mathé, C., Sagot, M.-F., Schiex, T., and Rouzé, P. (2002). Current methods of gene prediction, their strengths and weaknesses. *Nucleic Acids Res*, 30(19):4103–17.

[Mathe et al., 2006] Mathe, E., Olivier, M., Kato, S., Ishioka, C., Hainaut, P., and Tavtigian, S. (2006). Computational approaches for predicting the biological effect of p53 missense mutations: a comparison of three sequence analysis based methods. *Nucleic Acids Research*, 34(5).

[McLachlan et al., 2006] McLachlan, G. J., Bean, R. W., and B, J. L. (2006). A simple implementation of a normal mixture approach to differentical gene expression in multiclass microarrays. *Bioinformatics*, 22(13):1608–1615.

[McLachlan and Peel, 2000] McLachlan, G. J. and Peel, D. (2000). *Finite Mixture Models*. Wiley, New York.

[McLachlan et al., 1999] McLachlan, G. J., Peel, D., Basford, K. E., and Adams, P. (1999). The Emmix software for the fitting of mixtures of normal and t-components. *Journal of Statistical Software*, 4(2).

[Meilijson, 1989] Meilijson, I. (1989). A fast improvement to the EM algorithm on its own terms. *J R Statist Soc B*, 51(1):127–138.

[Mengersen et al., 2011] Mengersen, K. L., Robert, C. P., and Titterington, D. M., editors (2011). *Mixtures: Estimation and Applications*. John Wiley and Sons.

[Mooney, 2005] Mooney, S. (2005). Bioinformatics approaches and resources for single nucleotide polymorphism functional analysis. *Brief Bioinform*, 6(1):44–56.

[Ng and Henikoff, 2001] Ng, P. and Henikoff, S. (2001). Predicting deleterious amino acid substitutions. *Genome Research*, 11(5):863–874.

[Ng and Henikoff, 2002] Ng, P. and Henikoff, S. (2002). Accounting for human polymorphisms predicted to affect protein function. *Genome Research*, 12(3).

[Ng and Henikoff, 2006] Ng, P. and Henikoff, S. (2006). Predicting the effects of amino acid substitutions on protein function. *Annual Review of Genomics and Human Genetics*, 7(1):61–80.

[Ng et al., 2009] Ng, S., Turner, E., Robertson, P., Flygare, S., Bigham, A., Lee, C., Shaffer, T., Wong, M., Bhattacharjee, A., Eichler, E., Bamshad, M., Nickerson, D., and Shendure, J. (2009). Targeted capture and massively parallel sequencing of 12 human exomes. *Nature*, advance online publication(7261):272–276.

[Ng et al., 2010] Ng, S. B., Buckingham, K. J., Lee, C., Bigham, A. W., Tabor, H. K., Dent, K. M., Huff, C. D., Shannon, P. T., Jabs, E. W., Nickerson, D. A., Shendure, J., and Bamshad, M. J. (2010). Exome sequencing identifies the cause of a mendelian disorder. *Nat Genet*, 42(1):30–5.

[Oakes, 1999] Oakes, D. (1999). Direct calculation of the information matrix via the EM algorithm. *J R Statist Soc B*, 61:479–482.

[Olivier et al., 2002] Olivier, M., Eeles, R., Hollstein, M., Khan, M. A., Harris, C. C., and Hainaut, P. (2002). The IARC TP53 database: new online mutation analysis and recommendations to users. *Hum Mutat*, 19(6):607–14.

[Otis et al., 1978] Otis, D., Burnham, K., White, G., and Anderson, D. (1978). Statistical inference from capture data on closed animal populations. *Wildl Monogr*, 62:3–135.

[Pearson, 1894] Pearson, K. (1894). Contributions to the theory of mathematical evolution. *Philos Trans R Soc Lond A*, 185:71–110.

[Pepe, 2004] Pepe, M. (2004). *The Statistical Evaluation Of Medical Tests For Classification And Prediction*. Oxford University Press.

[Permuter et al., 2003] Permuter, H., Francos, J., and Jermyn, I. H. (2003). Gaussian mixture models of texture and colour for image database retrieval. In *Proceedings (ICASSP '03)*.

[Pollock, 1982] Pollock, K. H. (1982). A capture-recapture design robust to unequal probability of capture. *J Wildl Manage*, 46:751–757.

[Pritchard, 2001] Pritchard, J. K. (2001). Are rare variants responsible for susceptibility to complex diseases? *Am J Hum Genet*, 69(1):124–37.

[Pruitt et al., 2009] Pruitt, K. D., Harrow, J., Harte, R. A., Wallin, C., Diekhans, M., Maglott, D. R., Searle, S., Farrell, C. M., Loveland, J. E., Ruef, B. J., Hart,

E., Suner, M.-M., Landrum, M. J., Aken, B., Ayling, S., Baertsch, R., Fernandez-Banet, J., Cherry, J. L., Curwen, V., Dicuccio, M., Kellis, M., Lee, J., Lin, M. F., Schuster, M., Shkeda, A., Amid, C., Brown, G., Dukhanina, O., Frankish, A., Hart, J., Maidak, B. L., Mudge, J., Murphy, M. R., Murphy, T., Rajan, J., Rajput, B., Riddick, L. D., Snow, C., Steward, C., Webb, D., Weber, J. A., Wilming, L., Wu, W., Birney, E., Haussler, D., Hubbard, T., Ostell, J., Durbin, R., and Lipman, D. (2009). The consensus coding sequence (CCDS) project: Identifying a common protein-coding gene set for the human and mouse genomes. *Genome Res*, 19(7):1316–23.

[Qi et al., 2007] Qi, Y., Paisley, J. W., and Carin, L. (2007). Music analysis using hidden Markov mixture models. *IEEE Transactions on Signal Processing*, 55(11):5209–5224.

[R Core Team, 2012] R Core Team (2012). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

[Rabiner, 1989] Rabiner, L. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

[Ramensky et al., 2002] Ramensky, V., Bork, P., and Sunyaev, S. (2002). Human non-synonymous SNPs: server and survey. *Nucleic acids research.*, 30(17):3894–3900.

[Ramesh and Wilpon, 1992] Ramesh, P. and Wilpon, J. G. (1992). Modeling state durations in hidden Markov models for automatic speech recognition. In *ICASSP*, volume 1, pages 381–384. 1992 IEEE International Conference.

[Rao et al., 1977] Rao, D. C., Morton, N. E., Lindsten, J., Hultén, M., and Yee, S. (1977). A mapping function for man. *Hum Hered*, 27(2):99–104.

[Rattanasiri et al., 2004] Rattanasiri, S., Böhning, D., Rojanavipart, P., and Athipanyakom, S. (2004). A mixture model application in disease mapping of malaria. *Southeast Asian J Trop Med Public Health*, 35(1):38–47.

[Reich and Lander, 2001] Reich, D. E. and Lander, E. S. (2001). On the allelic spectrum of human disease. *Trends Genet*, 17(9):502–10.

[Reva et al., 2007] Reva, B., Antipin, Y., and Sander, C. (2007). Determinants of protein function revealed by combinatorial entropy optimization. *Genome Biology*, 8.

[Reva et al., 2011] Reva, B., Antipin, Y., and Sander, C. (2011). Predicting the functional impact of protein mutations: application to cancer genomics. *Nucleic Acids Res.*

[Reynolds and Rose, 1995] Reynolds, D. A. and Rose, R. C. (1995). Robust text-independent speaker identification using gaussian mixture speaker models. *IEEE Transactions on Speech and Audio Processing*, 3(1):72–83.

[Risch and Merikangas, 1996] Risch, N. and Merikangas, K. (1996). The future of genetic studies of complex human diseases. *Science*, 273(5281):1516–1517.

[Robinson et al., 2011] Robinson, P. N., Krawitz, P., and Mundlos, S. (2011). Strategies for exome and genome sequence data analysis in disease-gene discovery projects. *Clin Genet*, 80(2):127–32.

[Rödelsperger et al., 2011] Rödelsperger, C., Krawitz, P., Bauer, S., Hecht, J., Bigham, A. W., Bamshad, M., de Condor, B. J., Schweiger, M. R., and Robinson, P. N. (2011). Identity-by-descent filtering of exome sequence data for disease-gene identification in autosomal recessive disorders. *Bioinformatics*, 27(6):829–36.

[Romberg et al., 2001] Romberg, J. K., Choi, H., and Baraniuk, R. G. (2001). Bayesian tree-structured image modeling using wavelet-domain hidden Markov models. *IEEE Transactions on Image Processing*, 10(7):1056–1068.

[Schlattmann, 1996] Schlattmann, P. (1996). The computer package DismapWin. *Statistics in Medicine*, 15:931.

[Schlattmann, 2000] Schlattmann, P. (2000). Mixture models and modeling heterogeneity of the regional distribution of avoidable death in germany 1995. *Stud Health Technol Inform*, 77:417–422.

[Schlattmann, 2009] Schlattmann, P. (2009). *Medical applications of finite mixture models*. Springer, Berlin.

[Schlattmann and Böhning, 1993] Schlattmann, P. and Böhning, D. (1993). Mixture models and disease mapping. *Stat Med*, 12(19-20):1943–50.

[Schneider et al., 1986] Schneider, T. D., Stormo, G. D., Gold, L., and A, E. (1986). Information content of binding sites on nucleotide sequences. *J Mol Biol*, 188:415–431.

[Schork et al., 2009] Schork, N. J., Murray, S. S., Frazer, K. A., and Topol, E. J. (2009). Common vs. rare allele hypotheses for complex diseases. *Curr Opin Genet Dev*, 19(3):212–9.

[Scott, 2002] Scott, S. (2002). Bayesian methods for hidden Markov models: Recursive computing in the 21st century. *Journal of the American Statistical Association*, 97(457):337–351.

[Shao, 2003] Shao, J. (2003). *Mathematical Statistics*. Springer New York, second edition.

[Sifrim et al., 2012] Sifrim, A., Van Houdt, J. K., Tranchevent, L.-C., Nowakowska, B., Sakai, R., Pavlopoulos, G. A., Devriendt, K., Vermeesch, J. R., Moreau, Y., and Aerts, J. (2012). Annotate-it: a Swiss-knife approach to annotation, analysis and interpretation of single nucleotide variation in human disease. *Genome Med*, 4(9):73.

[Sigrist et al., 2010] Sigrist, C. J. A., Cerutti, L., de Castro, E., Langendijk-Genevaux, P. S., Bulliard, V., Bairoch, A., and Hulo, N. (2010). Prosite, a protein domain database for functional characterization and annotation. *Nucleic Acids Res*, 38(Database issue):D161–6.

[Sim et al., 2012] Sim, N.-L., Kumar, P., Hu, J., Henikoff, S., Schneider, G., and Ng, P. C. (2012). SIFT web server: predicting effects of amino acid substitutions on proteins. *Nucleic Acids Res*, 40:W452–W457.

[Sing et al., 2005] Sing, T., Sander, O., Beerenwinkel, N., and Lengauer, T. (2005). ROCR: visualizing classifier performance in R. *Bioinformatics*, 21(20):3940–1.

[Singleton et al., 2010] Singleton, A. B., Hardy, J., Traynor, B. J., and Houlden, H. (2010). Towards a complete resolution of the genetic architecture of disease. *Trends Genet*, 26(10):438–42.

[Stultz et al., 1993] Stultz, C. M., White, J. V., and Smith, T. F. (1993). Structural analysis based on state-space modeling. *Protein Sci*, 2(3):305–14.

[Sturt, 1976] Sturt, E. (1976). A mapping function for human chromosomes. *Ann Hum Genet*, 40(2):147–63.

[Sunyaev, 2012] Sunyaev, S. R. (2012). Inferring causality and functional significance of human coding DNA variants. *Hum Mol Genet*, 21(R1):R10–7.

[Szabo et al., 2000] Szabo, C., Masiello, A., Ryan, J. F., and Brody, L. C. (2000). The breast cancer information core: database design, structure, and scope. *Hum Mutat*, 16(2):123–31.

[Tavtigian et al., 2008a] Tavtigian, S. V., Byrnes, G. B., Goldgar, D. E., and Thomas, A. (2008a). Classification of rare missense substitutions, using risk surfaces, with genetic- and molecular-epidemiology applications. *Human Mutation*, 29(11):1342–1354.

[Tavtigian et al., 2008b] Tavtigian, S. V., Greenblatt, M. S., Lesueur, F., Byrnes, G. B., and the IARC Unclassified Genetic Variants Working Group, f. (2008b). In silico analysis of missense substitutions using sequence-alignment based methods. *Human Mutation*, 29(11):1327–1336.

[Thede and Harper, 1999] Thede, S. M. and Harper, M. P. (1999). A second-order hidden Markov model for part-of-speech tagging. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, ACL '99, pages 175–182, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Thompson et al., 2013] Thompson, B. A., Greenblatt, M. S., Vallee, M. P., Herkert, J. C., Tessereau, C., Young, E. L., Adzhubey, I. A., Li, B., Bell, R., Feng, B., Mooney, S. D., Radivojac, P., Sunyaev, S. R., Frebourg, T., Hofstra, R. M. W., Sijmons, R. H., Boucher, K., Thomas, A., Goldgar, D. E., Spurdle, A. B., and Tavtigian, S. V. (2013). Calibration of multiple in silico tools for predicting pathogenicity of mismatch repair gene missense substitutions. *Hum Mutat*, 34(1):255–65.

[Thorne et al., 1996] Thorne, J. L., Goldman, N., and Jones, D. T. (1996). Combining protein evolution and secondary structure. *Mol Biol Evol*, 13(5):666–73.

[Thusberg et al., 2011] Thusberg, J., Olatubosun, A., and Vihinen, M. (2011). Performance of mutation pathogenicity prediction methods on missense variants. *Hum Mutat*, 32(4):358–68.

[Thusberg and Vihinen, 2009] Thusberg, J. and Vihinen, M. (2009). Pathogenic or not? and if so, then how? studying the effects of missense mutations using bioinformatics methods. *Human Mutation*, 30(5):703–714.

[Wald, 1943] Wald, A. (1943). Tests of statistical hypotheses concerning several parameters when the number of observations is large. *Trans Amer Math Soc*, 54:426–482.

[Wang et al., 2006] Wang, H., Lin, C.-H., Service, S., Chen, Y., Freimer, N., Sabatti, C., and International Collaborative Group on Isolated Populations (2006). Linkage disequilibrium and haplotype homozygosity in population samples genotyped at a high marker density. *Hum Hered*, 62(4):175–89.

[Wang et al., 2010] Wang, K., Li, M., and Hakonarson, H. (2010). ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data. *Nu-*

cleic Acids Res, 38(16):e164.

[Wang et al., 2004] Wang, Z., Chen, Y., and Li, Y. (2004). A brief review of computational gene prediction methods. *Genomics Proteomics Bioinformatics*, 2(4):216–21.

[Watson and Chung Tsoi, 1992] Watson, B. and Chung Tsoi, A. (1992). Second order hidden markov models for speech recognition. In *Second order Hidden Markov Models for speech recognition*, pages 146–151. Fourth Australian International Confernce on Speech Science and Technology.

[Weldon, 1893] Weldon, W. F. R. (1893). On certain correlated variations in carcinus moenas. *Proceedings of the Royal Society of London*, 54:318–329.

[Wong et al., 2011] Wong, W. C., Kim, D., Carter, H., Diekhans, M., Ryan, M. C., and Karchin, R. (2011). CHASM and SNVBox: toolkit for detecting biologically important single nucleotide mutations in cancer. *Bioinformatics*, 27(15):2147–8.

[Yandell et al., 2011] Yandell, M., Huff, C., Hu, H., Singleton, M., Moore, B., Xing, J., Jorde, L. B., and Reese, M. G. (2011). A probabilistic disease-gene finder for personal genomes. *Genome Res*, 21(9):1529–42.

[Zelinka and Sigmund, 2010] Zelinka, P. and Sigmund, M. (2010). Hierarchical classification tree modeling of nonstationary noise for robust speech recognition. *Information Technology and Control*, 39(3):202–210.